電腦時代 COMPUTING AGE

18

工商業●教育●家庭娛樂●私人應用電腦雜誌　港幣十元



程式磁碟
各電視服務站有售

MD2D MINI-FLOPPY DISK

資宝牌 軟磁盤



* *High quality mini-floppy disk for data storage and retrieval.*
* *100% error-free of disk surface ensured by comprehensive tests.*
* *Prolongs drive read/write heads by its superior surface smoothness.*
* *Best selection of jacket optimize drive characteristic.*

Item Model No.	MD1D	MD2D
Disk Size (Inch)	5¼	5¼
Sector	Soft	Soft
Usable Side	Single	Double
Density	Double	Double
Tracks per inch (TPI)	48	48
Number of Tracks	40	80
Storage Capacity (Bytes)	163,840	327,680

Super 5

A & N ENTERPRISE CO., LTD.
20/F WESTERN CENTRE.,
40-50 DES VOEUX ROAD, WEST, H K
TEL: 5-461179 5-488463
TELEX: 65218 MCLWW HX
FAX: 5-497149

Super 5

DOT MATRIX PRINTER
EN-P1091/EN-P1090
点阵式打印机

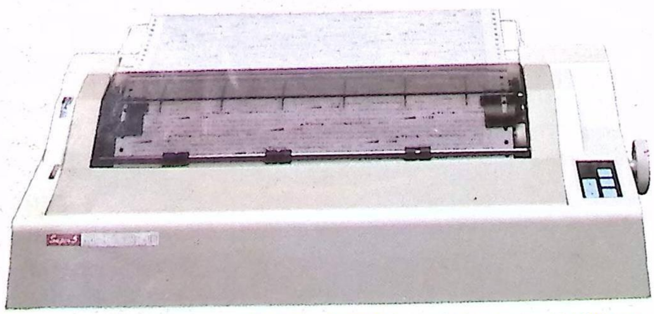
NEW VERSION
EP—1201



EN-P1091
- Print Mode Selector Switch
- Near-letter quality printing
- Print Speed 120 cps.

EN-P1090
- High resolution, dot-addressable graphics
- Print speed 80 cps.

100% IBM* PC COMPATIBLE



SUPER 5 MBE-420 15" PRINTER • Print speed 180 cps.



SUPER 5 EP-1807 15" 7 COLOR PRINTER
- Inked platens printer

NEW

SUPER 5 CP80/CPA80 9 pin printer

SUPER 5 MONITOR FOR IBM PC 资料显示器

SUPER 5 POWER SUPPLY 变压器
for IBM* pc Max. output Power 133.9W

SUPER 5 5.25" Floppy Diskette — EE50(SSDD)/EN-320K (DSDD)

SUPER T-40e/C-40 DISK DRIVE 软盘磁碟驱动器

Plus FDD for IBM* PC

CHINON 502 / EPSON SP-521 / TEAC FD-55B / MATSUSHITA JA551-2

* IBM PC is a registered trademark of International Business Machine Corp.

Super 5

EI-EN ELECTRONICS, LTD.
22-23-213 Ebisu-I-Chome Shibuya-Ku, Tokyo, Japan.
Telex J23325 EIEN ENT FAX (03) 446-6530.

A & N ENTERPRISE CO., LTD.
20/F., Western Centre, 40-50 Des Voeux Rd., West, H.K.
Tel: 5-461179 (3 lines) 5-488463 Telex: 65218 MCLWW HX.

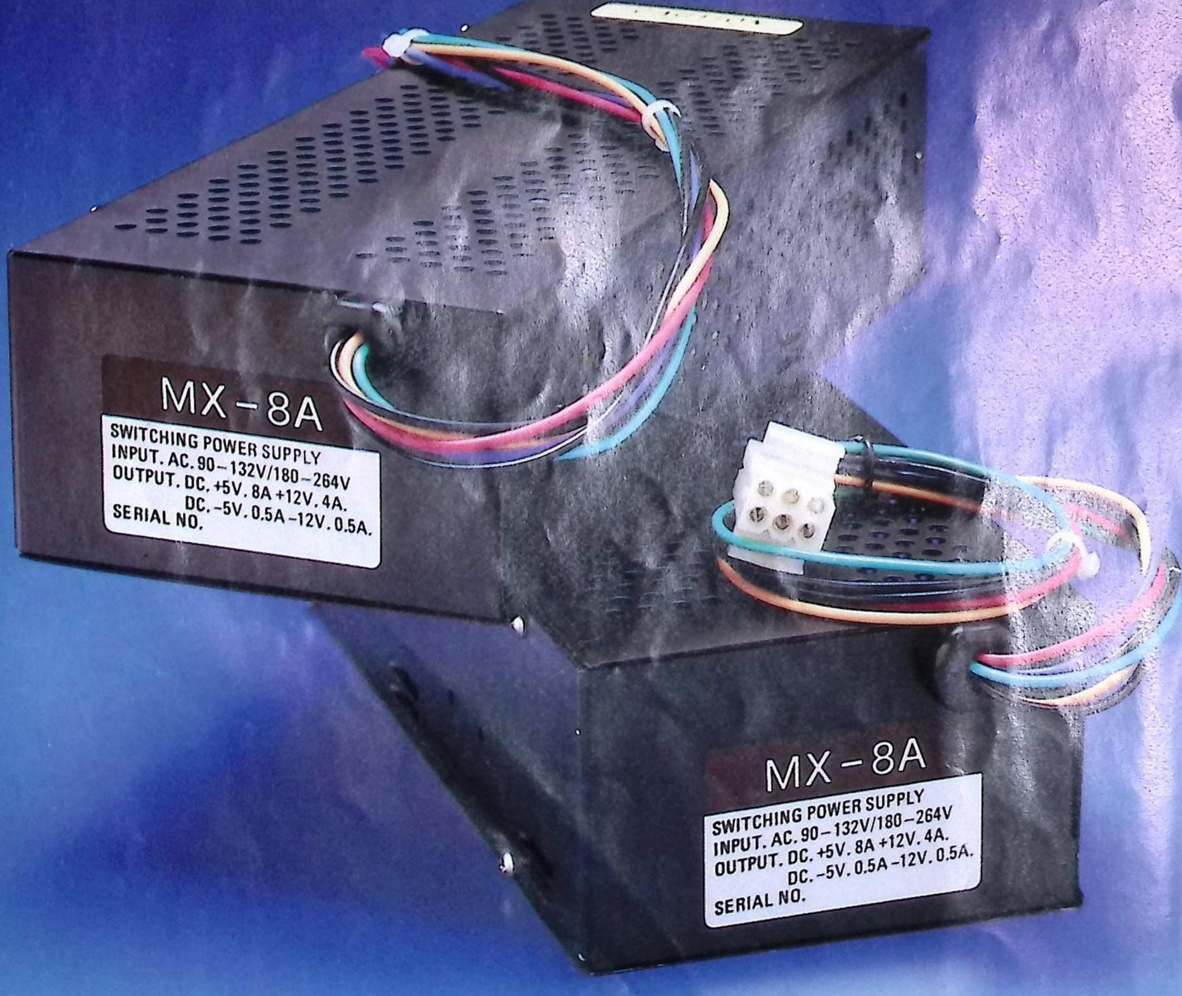
超級電源供應器

MX-8A

勁力十足・令主機如虎添翼

10小時動態測試・短路保護試驗・一年免費保養

歡迎出口或批發
敬希垂詢



適用於AP及各種單板機，最高負載如下：

INPUT AC・90-132V/180-264V
OUTPUT DC.・+5V・8A+12V・4A
DC −5V・0.5A−12V・0.5A

香港電工器材
電話： 3-876673

COMPUSOFT,

一本蘋果電腦用家必備之電腦雜誌

磁碟月刊經已面市



現已在各大電視服務站公開發售

COMPUSOFT 磁碟月刊，每期均出版兩張 5 ¼″雙面灌錄的磁碟(合共四面)，以DOS 3.3 FORMAT，適用於任何蘋果電腦或相容的機種。

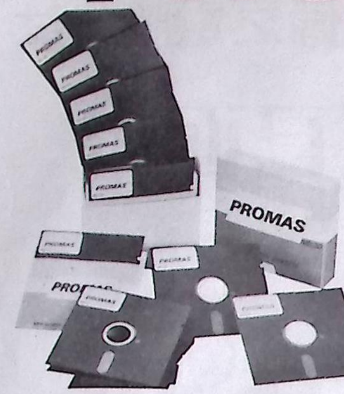
COMPUSOFT 磁碟月刊每期均包羅有極富娛樂、刺激兼並的電腦遊戲。教育軟件、適合任何行業的實用工具程式、電腦自學文章，最新軟件測試報告……等精彩內容，是一本適合電腦發燒友、未來電腦從業員、學校教職員，學生，家長及有志投身電腦軟件工業人士的必備電腦雜誌。

磁碟月刊每期所包含內容的技巧，更是透過一羣專業程序設計師精心泡製的成果，它有着極大的研究價值，是用家們值得珍藏之電腦雜誌。

由六月開始，COMPUSOFT 磁碟月刊均可在各地鐵或九廣鐵路電視服務站有售。

購備COMPUSOFT 磁碟月刊，只花小小代價，獲多多的得益。

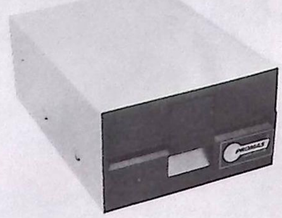
PROMAS 電腦產品



5.25″磁碟
雙面雙倍密度，儲存量達 500 K ，適合一般電腦使用。全日本製造。



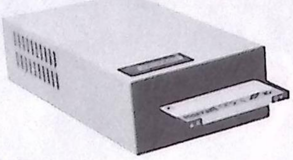
MODEL JT 5.25″薄身磁碟機
多種「WRITE PROTECT」及「HEAD SEEK SPEED」選擇，適合不同應用。APPLE II*及相容電腦專用。全日本製造



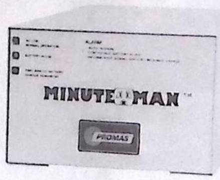
AS-100 5.25″磁碟機
傳統設計，品質檢定合格，APPLE II*及相容電腦專用。全英國製造。



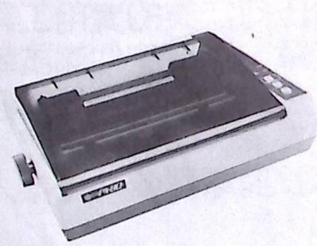
MODEL JA 5.25″磁碟機
全新設計機械部份，特別柔順。APPLE II*及相容電腦專用。全日本製造。



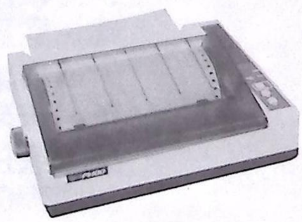
305 3″磁碟機
多重保護資料，與一般APPLE II* 磁碟機共用或可單獨使用。全日本製造。



MINUTEMAN後備電源供應器
防止一般電源問題對電腦及精密儀器造成之破壞。全日本製造。



PX-80印字機
印字速度 每秒80字。CENTRONICS介面接口，適合一般電腦使用。全日本製造。



PX-100印字機
印字速度 每秒 100 字，功能與P X -80相同。全日本製造。



PF-150 手提圖文傳眞機
攜帶方便，可與世界各地傳眞機互傳訊息。全日本製造。

總代理：

PROMAS COMPUTER CENTRE LTD.

寶馬士電腦有限公司

九龍尖沙咀柯士甸路22-26號好兆年行608室　電話：3-674687　電訊：HX 38316

PROMAS

打印機



PX-80點矩陣式打印機

- 來回方式印字，每秒可印80字。
- 採用新設計印字頭(PRINT HEAD)，高密度式印字。
- 每行可印71－142字。
- 可用普通紙及電腦紙。
- 墨帶壽命特長。
- CENTRONICS平行式介面接口，適合一般電腦使用。
- 全日本製造。



PX-100點矩陣式打印機

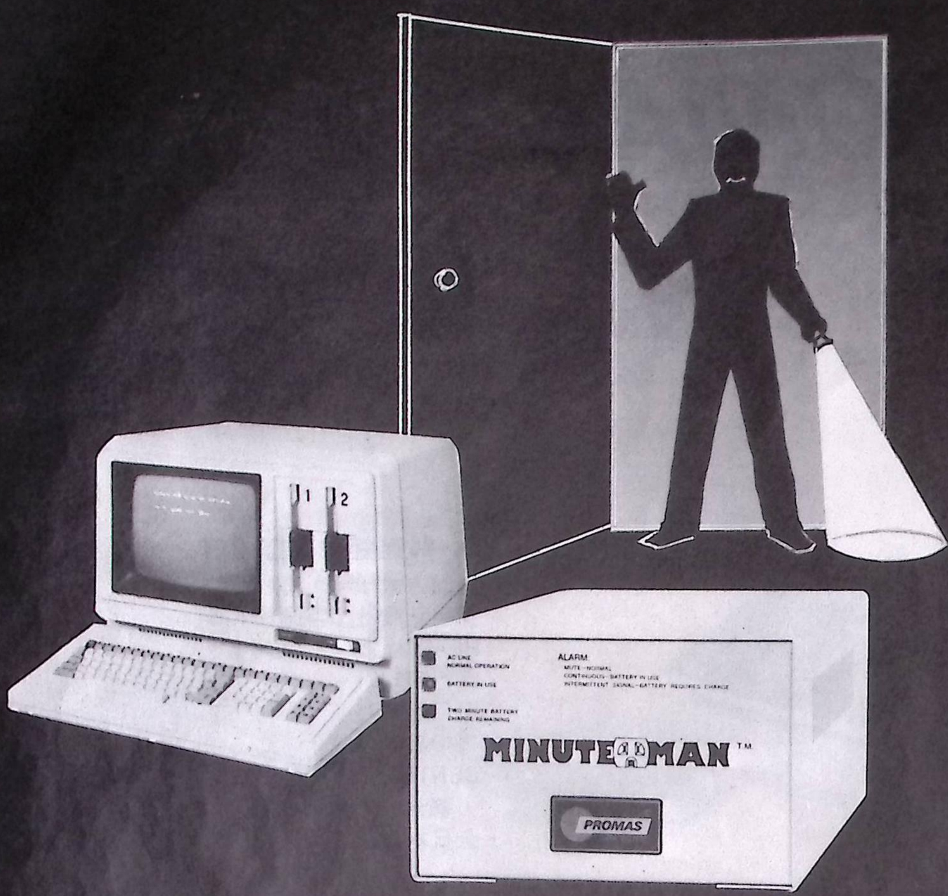
- 高速度來回式印字，每秒可印100字。
- 功能與PX-80相同。
- 全日本製造。

總代理：

PROMAS COMPUTER CENTRE LTD.

寶馬士電腦有限公司

九龍尖沙咀柯士甸路22-26號好兆年行608室　電話：3-674687　電訊：HX 38316



停電 都冇損失！

全新設計線路
效果更佳更理想

Minuteman UPS「自動後備供電器」

保障電腦及電子器材運行正常，不受電源中斷影響而失去寶貴資料及損壞昂貴器材。

- 可用作防止電壓下降、電壓冲擊及電源中斷等，確保電源供應正常。
- 當市電停止供應時，能立即自動提供後備電源，並以蜂鳴器警報及指示燈顯示。
- 供電時間視乎配用器材之耗電量而定。
- 在市電供應恢復正常後，本機會隨即自動充電。
- 適用於任何電腦，硬碟儲存系統及精密電子儀器。
- 全日本製造，價錢比同類型產品便宜一半。

總代理：

PROMAS COMPUTER CENTRE LTD.

寶馬士電腦有限公司

九龍尖沙咀柯士甸路22-26號好兆年行608室　電話：3-674687　電訊：HX 38316


新設計磁碟機
PROMAS Model JT
PROMAS
MARK II
WRITE PROTECTION
Write thru write-protect seal
Normal write-protected mode
All disks write-protected
HEAD SEEK
Default seek
Slow seek
DEFAULTS:
● 獨特"SLOW SEEK"功能，存取資料更準確。
● 內置WRITE PROTECT及HEAD-SEEK速度選擇掣。
● 採用新設計高質素機械部份，可靠程度更高。
● 支架用純不銹鋼製造。
● 完全適用於APPLE II*系列電腦。
● 全日本製造及質素檢定合格。
總代理：
PROMAS COMPUTER CENTRE LTD.
PROMAS 寶馬士電腦有限公司
九龍尖沙咀柯士甸路22-26號好兆年行608室
電話：3-674687
電訊：HX 38316
*Trademark of Apple Computer Inc.

PROMAS PX-136

市面最經濟之15吋高速打印機

全新



- 速度每秒高達130字。
- 內置獨有IBM[1]專用軟件ROM與IBM PC[1]電腦軟件完全相容。
- 可印多國文字，IBM[1]圖案文字及BIT IMAGE圖案。
- 多種印字功能，每行可印68-233個大小文字。
- 獨特新設計之「雙軌式墨帶」，可使墨帶壽命加長一倍，合乎經濟原則。
- 標準CENTRONICS平行式界面接口，適合一般電腦、APPLE[2]、IBM PC[1]及同類型電腦使用。
- 全日本製造。一年代理商免費保用。

享有代理商保證
才是專業性保用

1 Registered trademark of International Business Machine Corp.
2 Registered trademark of Apple Computer Inc.

總代理：
PROMAS COMPUTER CENTRE LTD.
寶馬士電腦有限公司
九龍尖沙咀柯士甸路22－26號好兆年行608室　電話：3-674687　電訊：HX38316



★ 學習尖端科技

★ 邁向成功坦途

現代電腦系統訓練中心

MODERN COMPUTER SYSTEM TRAINING CENTRE

—— 暑期電腦課程招生 ——

COURSE 課程表	上課編制	時數	學費
電腦語言課程			
(1) BASIC PROGRAMMING COURSE	個別或班制	16HRS	$280
(2) ADVANCED BASIC COURSE	個別或班制	15HRS	$300
(3) ASSEMBLER LANGUAGE COURSE	個別或班制	16HRS	$350
(4) "C" LANGUAGE COURSE	班制上課	16HRS	$800
(5) 兒童電腦班(LOGO及初級BASIC)	班制上課	10HRS	$200
電腦硬件課程			
(1) 6502電腦自嵌實習班(軟硬件理論兼備)	個別或班制	30HRS	$500
(2) APPLE-II電腦修理班	班制上課	80HRS	$1500
(3) IBM-PC電腦修理班	班制上課	80HRS	$1800
綜合應用課程			
(1) dBASE-II入門及商業應用初級及高級班	個別或班制	10HRS	$300
(2) CP/M操作系統	個別或班制	16HRS	$350
(3) DOS應用及拆碟班	班制上課	20HRS	$550
(4) WORDSTAR TRAINING COURSE	個別或班制	16HRS	$350
(5) PEACHTREE A/C SYSTEM會計系統	個別或班制	16HRS	$800

上述學費概以班制上課為準，個別教授另議

創辦叁年　電腦碩士任教

本校師資　(1)香港及中文大學電腦系，英美加電腦系畢業生。　(2)英國曼徹斯特大學電腦碩士，英國電腦學會會員。

本中心開放時間：10：00A.M—10：00P.M

地址：九龍深水埗福華街156號黃金商場地庫20號室，54號室及67號室。

電話：3－601531　3－617206

香港教育司署立案（1957年創校）

世界函授學校

規模大　優點多　科目好

講義
全部經教育司署審定，閱讀容易，印刷精美，免費供應。

導師
國內外著名大學學士，經香港教育司考核合格，教導認真，決不馬虎。

費用
除低廉之學費外，學生決無額外負擔。

文憑
本校頒發文憑，均經政府簽証，實是最佳學歷証件。

校址：
香港九龍彌敦道五六六號9樓A座
電話：③—803532

初級電腦科
電腦基本原理，電腦操作，邏輯電路，程式編寫，八個月畢業。

初級無綫電科
全部圖解教導，原理裝修並重，極易學習，十個月畢業。

半導體裝修科
配合進步的技術，具初學或職業的裝修人員必修的學科，十個月畢業。

實用數理
可以加强數理基礎，費用少，收效大，6個月畢業。

x+y=z

實用商科
著重實際應用，加强商業學識，期間短，學習易，六個月畢業。

無綫電工程
設計，研究和深造的理想學科。一年半畢業。

汽車工程
適合職業及業餘人員加强對汽車方面之學識。

冷氣工程
冷氣，冷藏人員缺乏，專門學校極少，一年半畢業。

商業美術
內容新，範圍廣，包括美術和最新廣告設計。一年畢業。

應用漫畫
漫畫可訓練頭腦，增强思想，取材新，學習易，六個月畢業。

INK

本港學生可利用電話詢問索取章程　電話③－803532

請依綫剪下，填好姓名，回郵地址，貼在信封上投寄，本校即免費寄上簡章

郵件請寄：P.O. BOX 70011, KOWLOON, CENTRAL POST OFFICE
九龍中央郵政局信箱 70011 號

技術就是財產

（寄自）
姓名：
地址：
（請註明要學的科目）
- □初級電腦科
- □無綫電工程科
- □初級無綫電科
- □半導體裝修科
- □實用數理科
- □汽車工程
- □冷氣工程
- □實用商科
- □商業美術
- □應用漫畫

請貼上郵票

香港教育司署立案

To：世界函授學校註册處
P. O. BOX No. 70011
KOWLOON CENTRAL POST OFFICE,
HONG KONG.

香港九龍中央郵局信箱70011號

電話：K-803532

學問就是力量

Apple
迷大喜訊

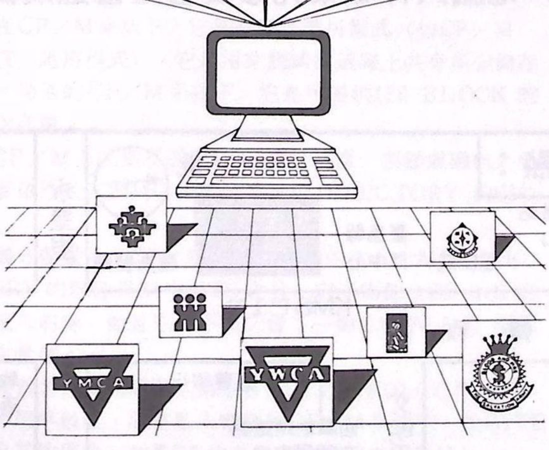
精美電子公司推出兩大皇牌
Mocking Card 及 *Track Reader*

Mocking Card ■ 使音響效果方面達到現場一樣，有眞實感，尤其是配合 Sky Fox 及 Music Construction Set 與街機毫不遜色。現時可配合用之 Software 已經非常之多，將來且有更多的軟件支持，其它同類卡無法相比。
（原裝線路）

Track Reader ■ 磁碟讀軌器，能讀出½ Track，可看到磁頭走動之情況，用以配合抄碟，更可達到最佳效果及可以減少抄碟的時間。

精美電子公司　JAN MI ELECTRONIC CO.
九龍深水埗福華街黃金商場地庫48號A
Shop 48A, Basement, 146-152 Fuk Wa Street, Golden Center, Golden Building Kowloon.　Tel: 3-7289086

電腦證書課程



香港區：	Y.W.C.A.女青年會總會所	5−223101 內線10
	Y.M.C.A.青年會黃傳經中心	5−671571
	香港青年協會民新街中心	5−641277
柴灣區：	循道衛理愛華服務中心	5−580108
灣仔區：	浸會愛羣社會服務處	5−724365
香港仔區：	香港青年協會黃竹坑中心	5−538850
旺角區：	救世軍九龍中央堂	3−312555
深水埗區：	澤安青年中心	3−7763312
黃大仙區：	救世軍啓德隊（業餘進修部）	3−223825
紅磡區：	聖公會聖匠社區中心	3−620301−8
葵涌區：	Y.M.C.A.新界會所	0−200266
荃灣區：	Y.W.C.A.荃灣社會服務處	0−402863
沙田區：	沙角青少年中心	0−6478816
	新翠青年中心	0−6981922
屯門區：	Y.M.C.A.屯門會所	0−848171

Y.W.C.A.

香港基督教女青年會
總會所：香港麥當奴道一號
5-223101內線10（IBM）

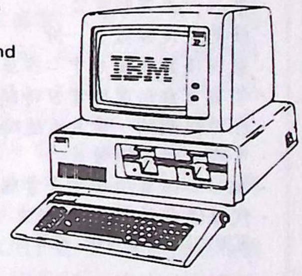
第三屆全日制
暑期電腦專業訓練課程

· 課程內容全面：

主　修：Introduction to Computer
Date Processing System
Computer Analysis
Computer Languages and Programming
Basic –Cobol –Pascal

專業訓練：DBASE II
Word Processing
Visicalc/Multiplan

副　修：Computer Marketing
Computer Accounting
File Handling
Human Relationship



· 歡迎中五至中七報名
· 全期兩個月（每月四佰二十元）

中華基督教青年會
窩打老道會所：九龍窩打老道二十三號
3-319111內線3050（SHARP）
黃傳經中心：西灣河街46-56號西灣河大厦二樓
5-671571（IBM）

Y.M.C.A.

專業電腦中心
PROFESSIONAL COMPUTER CENTRE

歷史悠久　深受推崇

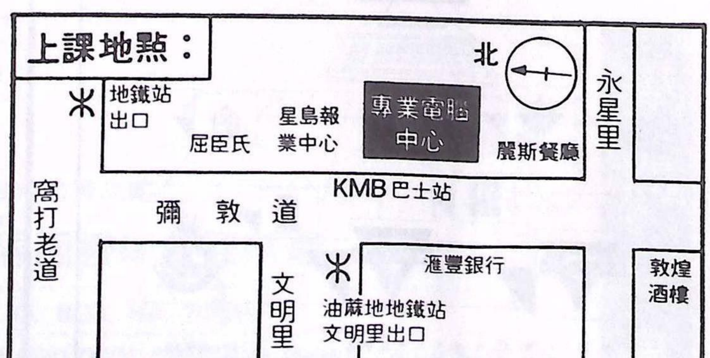
推廣電腦知識
贈閱半年電腦雜誌

親愛的學員：您們好。爲推廣電腦知識，凡於1985年8月30日之前參加下列任何一項課程，各學員均享有電腦雜誌贈閱半年的優惠。價值港幣陸拾圓，讓大家在進修期間，有良好的參考讀物

課　程：

1.基本電腦概念及程式設計
2.BASIC 程式設計
3.中學會考電腦科補習
4.6502電腦裝嵌、修理、職業先修課程
5.WORDSTAR 商業電腦打字課程
6.MULTIPLAN及VISICALC操作及應用
7.DBASEⅡ操作及應用
8.CP/M操作系統研究
9.Z-80及6502組合語言
10. 電腦入行職業先修實習證書速成課程

- 理論與實習並重
- 課程深入淺出，適合各階層人仕
- 每人一套電腦實習
- 結業後獲頒證書一份
- 成績優異者由本中心推薦加入電腦公司工作
- 學員可自由選擇實習時間
- 在課程期間，學員可隨時回校免費實習
- 可代公司訓練僱員
- 本中心備有中學會考電腦科參考資料
- 代購優惠服務（電腦雜誌　電腦軟件　磁碟　印字機　電腦文具　電腦書籍　磁碟機　電腦附件及各種名廠電腦）



索取章程，查詢及報名處：

專業電腦中心　九龍彌敦道486號益南華厦3樓電梯2字　電話：3-328248　3-318960

或

實發書業專門店　九龍深水埗欽州街94號高登商場一樓76A, B　電話：3-602326

四大磁碟系統磁碟資料存放格式詳析

DOS 3.3, CP／M, PASCAL, PRODOS

第二篇：CP／M 2.20系統

上期爲大家深入剖析DOS 3.3 系統下的磁碟儲存資料格式，今期，我們繼續討論另一種受到普遍用戶使用的系統——CP／M。

從圖(3)之中，大家可看到CP／M2.2 的磁碟系統列表，磁軌＄0至＄02是儲錄整個CP／M系統。不像DOS 3.3，CP／M 的檔案查閱磁軌是由磁軌＄3的前面0至7開始，也即是圖2的「D1—D8」（見上期同文）。

由磁軌＄3至＄22，圖2中是顯示了＄02至＄7F 的一BLOCK NO ，它是代表磁軌／扇面的參值。CP／M的儲存檔案方法是以1024BYTES 爲標準單位容量，相比256 BYTES一扇面的DOS 3.3，CP／M是將4個256 BYTES 的方塊合成1K長的扇面容量。CP／M的BLOCK＄00與＄01是保留給磁碟檔案名稱查閱，而其他資料檔案是可儲存於BLOCK ＄02至＄7F 上。整隻檔案磁鍵的儲存空間是可分成126 BLOCK，（每個BLOCK爲1024BYTE）或合共129,024 BYTE 。

在CP／M系統下，它是有很多應用程式（如CP／M STAT 應用程式），它是用來測試出磁碟上共有多少儲存空間。通常的CP／M系統下，它是可提供126 BLOCK 的資料儲存量。

CP／M系統是不需像DOS 3.3一樣，要顧慮磁軌／扇面的參值列表，因爲這些參值是儲於DIRECTORY 的磁軌內。

圖3便顯示了CP／M的檔案查閱的格式，每個DIRECTORY 的參值爲32 BYTE之長，它不需像DOS 3.3 的檔案讀入前奏，而且它亦不需記錄下一個CATALOG 扇面之所在參值。

檔案名稱查閱扇面的第1BYTE必爲＄00，它代表CP／M的用戶數號，而在私人電腦的CP／M系統下，此BYTE則作出其他用途，如＄E5代表檔案已被刪去的參值。

緊接其後的11個BYTES （此已跳越了8個BYTES長的檔案名）便是檔案的伸延檔名，如CP／M的PIP 應用程式，它的伸延檔名便爲.COM—亦即全寫爲PIP.COM，其中的豆號是由CP／M系統自動加上顯示於螢幕上。

在檔名伸延3個BYTES 後接的1個BYTE是用作區別不同的檔案查閱下才顯示出來。繼續的另外2個BYTES是CP／M 沒有提及，亦即爲未來的擴展用途。再下一個BYTE爲檔案數目，而餘下16個BYTES 便爲檔案的BLOCK NO 和位置。

檔案的BLOCK NO與位置是相似於DOS 3.3 的磁軌／扇面參值的功能。就以CP／M的應用程式PIP.COM 檔案的BLOCK NO與位置，其分別如下

02 03 04 05 06 07 08 09 00 00 00 00 00 00 00 00

以上的數據顯示了PIP.COM檔案是佔用了CP／M 8K（相等8BLOCK）的儲存量（02—09）。每個BLOCK位置的長度是被限制於16K BYTES 。假使一個長於16K BYTES 的檔案，CP／M又怎樣應付呢？

CP／M是會再重新產生一個新的檔案查閱參值表，其檔名亦沿用舊檔名，而參值中却加上了＄01於伸延表上。這是代表CP／M會將剩餘的資料檔案由BLOCK＄17儲錄至BLOCK ＄32上，而檔案的BLOCK NO便會＄17和＄32了。

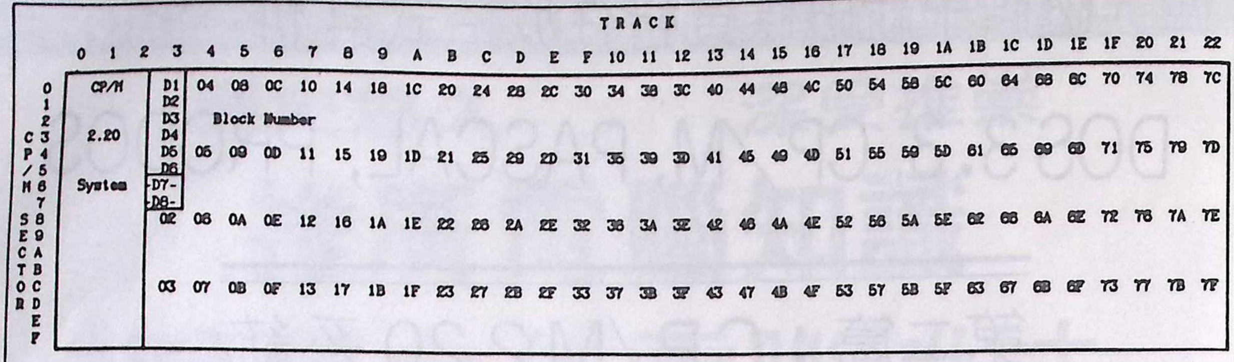
再參看圖3的參值，其首一個BYTE是代表記錄群數目（參值爲＄3A）。還有，CP／M是會將1BLOCK 再分爲8個（128 BYTE長）記錄群，而標準的CP／M 是每次祇讀寫128 BYTE的記錄群（此有別於DOS 3.3的256 BYTES）。又如CP／M的PIP.COM檔案是8BLOCK 的長度，所以CP／M在讀寫PIP.COM 檔案時，便要分爲64個記錄群。

剛才提出＄3A參值是指出某檔案共分成了＄3A個的記錄群。比方是CP／M的PIP.COM 檔案是佔用64個資料群中的58個而已，亦即相等於DOS 3.3 的32SECTOR中祇用了29個SECTOR—其中3個SECTOR是完全空白的。所以，我們可以看到CP／M 在儲存一個極短檔案（如1025 BYTE），它便要以2BLOCK （2K）的磁碟空間去儲存整個檔案。這樣，CP／M 便浪費接近1K的資料容量了。

理論上，CP／M的DIRECTORY是可以處理接近64個檔案名稱與參值，每個的長度爲16BLOCK（共1024BLOCK），剛巧相等APPLE DOS的4096個扇面的容量。但實際上，APPLE CP／M的5¼吋磁碟却祇提供126 BLOCK（相等504 SECTOR）而已。

由MICROSOFT替APPLE編寫的CP／M系統亦祇可提供48個檔案的處理，而同時佔用了8個中的6個DIRECTORY扇面。原則上，MICROSOFT 可以擴展APPLE

四大磁碟系統磁碟資料存放格式詳析



圖(3)　CP／M 磁碟分佈圖

D1－D6＝DIRECTORY BLOCK
00＆01，8FILE／SECTOR
6個SECTOR ×8FILE＝每張磁碟最多48個FILE
D7＆D8 永不使用（$E5）
所有檔案的存放均由最低的自由BLOCK開始，然後根據BLOCK NO. 的大小依次由小到大存放
所有BLOCK 的使用亦依SECTOR NO 的大小而定，由小到大。
例如：BLOCK 02 存放於TRACK 3，SECTOR 8，SECTOR 9，然後是A、B。
CP／M DIRECTORY ENTRY
沒有HEADER
FILE ENTRY，32 BYTE／FILE

```
0050495020202020434F4D0000003A020304050607080900000000000000
!-!P I P - - - - - -!C O M!-!- -!-!- - - - - - - - - - - - -!
^ ^          檔案名字           ^    ^ ^--       BLOCK 位置   ----------^
!                               !      ^--      FILE內RECORD數目
'                               !            1個RECORD＝128 BYTE
^---     用戶數目               !            1個BLOCK＝8個RECORD＝4個SECTOR＝1024BYTE
   (假如FILE被DELETE，會是E5)  ^--          EXTENT 數目，每個EXTENT有16個BLOCK， 即等
                                        如1個 DIRECTORY ENTRY
```

CP／M 檔案名稱的處理能力，但這可能MICROSOFT 要顧及40行顯示的APPLE 系統是很難同時顯示64個檔名，因此，你可發覺8個DIRECTORY 扇面內有2個扇面是被浪費的。

在圖3中，所有資料檔案是祇容許儲存於BLOCK ＄02－＄7F間（相等磁軌＄03至＄22），而磁軌＄00－＄02則保留爲CP／M 系統之用。APPLE CP／M 是提供了格式一片資料磁碟時，不需將CP／M 系統載入磁碟中。這樣，用者便可將磁軌＄00－＄02 用作資料儲存之用。這早期的CP／M2.20 版本是沒法讀取磁軌＄00至＄02 間的資料數據，但後期的CP／M2.23 便解決此項缺點。在CP／M 2.23下，磁軌＄00－＄03 是被看作CP／M的第＄80BL－OCK，而磁軌＄02的第＄0C－＄0F 的扇面則爲CP／M的第＄8 BLOCK。

通過CP／M STAT 應用程式是可以將某一個檔案的態度詳列出來，如PIP.COM的態度如下：

```
Rects  Bytes  Ext  Acc
 58     8k     1   R/W   B:PIP.COM
```

「REC 58」是代表PIP.COM 檔案共用了磁碟上共58個記錄群（約29扇面）；「BYTES 8K」則檔案共實際佔據了8 BLOCK 的空間；「EXT 1」表示檔案是少於16BLO－CK的範圍內。

CP／M的檔案寫入程序是依據BLOCK NO而遞增，而每當遇到已佔用的BLOCK時，CP／M便會跳越至其他空置的BLOCK 上，繼續寫入資料。這種方法是極相似APPLE DOS 3.3，但卻又別於APPLE PASCAL ——將一個檔案完全佔用一個或一堆連續的扇面上。現在，簡列出CP／M 2.20 的磁碟檔案系統格式如下：

(1)首先，讀入磁軌3的DIRECTORY扇面D1－D6 ，然後由CP／M LOGICAL扇面＄0移至扇面＄5上，再找尋所需檔名。假若你的ZAP程式與CP／M的扇面編排有所分別，這可參閱圖1見上期同文的扇面對換圖表。

(2)當你找到所需檔名時，由右邊數回15個BYTES 去找出檔案的BLOCK NO 。圖3與1是需要列出檔案於磁軌與扇面的所在處。

(3)繼續讀入其餘的DIRECTORY，直接找出最後的檔案爲止。若16 BLOCK 的檔案位置BLOCK 已被塡滿，則「EXT」需被置爲2（參看圖3）。

(4)讀取檔案的磁軌／扇面的次序與BLOCK NO 的翻譯列表。■

香港微型電腦訓練中心

教署立案（1980年）

招　生

(1)本校師資優良，課程由教育司署認可之電腦專業人仕任教。
(2)本校置有二十多部高速優良美國電腦系統。
(3)除正式授課及實習外，另提供自由時間免費實習電腦，加强訓練。
(4)每月均有新班，每星期上課一或兩次，時間分上、下午或晚班制。
(5)學費包括講義，電腦室實習，考試及證書費用。
(6)授課語言：中文（粵語），輔以英文名詞。

***本校備有齊全之各類商業軟件，方便學員在短期內熟習應用電腦。**

A. 短期電腦課程

(1)中學會考補習班	($150)一個月
(2)初級電腦班（概念＋操作＋BASIC 語言）	($340)25 小時
(3)ADVANCED－BASIC PROGRAMMING	($360)25 小時
(4)COBOL OR PASCAL	($420)30 小時
(5)ADVANCED －COBOL	($450)30 小時
(6)CP/M & MP/M (MULTI－USER) OPERATING SYSTEMS	($340)20 小時
(7)ASSEMBLER OR FORTRAN	($450)30 小時
(8)SUPERCALC OR MULTIPLAN	($360)12 小時
(9)WORD PROCESSING OR dBASE II	($450)15 小時

B. 一年制課程（代報公開試）：

(1)G.C.E. 'O' Level Examination Computing Studies
(2)British Computer Society Examination Studies

※詳情及開課日期，請向本中心校務處詢問※

校址：九龍佐敦道渡船街1－4號金霞閣 A座②字樓
（碼頭行人天橋側，大家樂鄰）
詢問電話： 3-858615 3-329996

美國DENNISON磁碟

磁碟中皇牌



- ■每隻磁碟均有出廠號碼紅印，保證100％美國製造。
- ■無限次數，終生保用，由DENNISON廠提供。
- ■溫度及濕度之耐力達到最高極限要求。
- ■零點誤差，資料提存，100％記憶系統保存。
- ■雷射式GAMMA FERRIC OXIDE磁粉掃描程序，確保碟面平滑如鏡，使磁頭與碟面接觸，輸出均勻調頻。
- ■DENNISON 獨有防靜電纖維內套，隔除靜電干擾，滿孔設計具有清潔作用，使磁碟生命長存。
- ■雙面四密度超高科技產品，適合IBM任何機型。
- ■全部製作不經人手，生產管理嚴格完善。超ANSI 及 ECMA專業要求。
- ■品種齊全，適合任何類型、大小電腦之要求。
- ■全美國銷量DENNISON 佔超過百分之十以上，並爲美國各大系統公司指定供應商。

總代理：FATTAFA

富力達國際有限公司
FATTAFA LIMITED

香港葵涌葵豐街1-15，盈業大厦12樓42號
No. 42, 12/F., Profit Ind Bldg., 1-15 Kwai Fung Crescent, Kwai Chung, Hong Kong.
TEL: 0-210110 (10 lines)
TELEX: 49531 FATFA
FAX: 0-252911

一張軟磁碟是怎樣誕生？

磁碟的製造過程詳論（上）

對任何軟性磁碟而言，製作方法都是相同的。基本上可以分爲兩個部分：第一是套的製造，而第二是磁性媒體的製作。在這裏我們首先向大家介紹磁性媒體的製作程序。爲了簡明起見，我們首先把製作過程的流程圖列出作一參考：

原材料的處理

↓

（磁粉）加瑪氧化鐵的複蓋

↓

表面的打磨過程

↓

第二次舖設磁性媒體的表面

↓

磁性媒體的測試

↓

磁性媒體成品

現在讓我們開始討論製作磁性媒體的過程。首先這些磁性媒體的原材料是一大卷長約三千呎的雲母片（美國杜邦化工DUPONT 廠的專利品），厚度是3mils（三千份之一吋），它的正確學名是BILATERALLY ORINTED POLYE—HTYLENE TERAPHTHALATE。 磁碟製造商在購入這些原材料後，便在其兩個表面之上以膠結系統（BINDER SYSTEM） 舖上一層加瑪氧化鐵（GAMMA FERRIC OXIDE)。通常製作磁性媒體的膠結系統都是POLYURET—HENE 方式的，不過亦有些廠商採用其它合成方法如SARAN、VINYLIDENE CHLORIDE來作膠結系統，使加瑪氧化鐵以交义連結或半交义連結方式舖在表面上。其實不論任何膠結方式都有其特別的優點和缺點，但最重要的是使磁碟機上的磁頭和磁性媒體兩者都能有較長的壽命。爲了這個原故，高質磁碟在其製作磁性媒體時會將加瑪氧化鐵舖在原材料的兩個表面上，使符合雙面雙倍密度的基本標準，然後在稍後討論的測試過程上，把各種等級的磁性媒體分別出來。這些一卷卷的磁性媒體在檢定合格之後便會被鑄模切爲8吋或5¼吋直徑的〝碟〞。與此同時廠商會依據客戶需求量將不同鑄模切出一定數量用作32硬磁區或10—16磁區的磁碟，並且在規格要求的位置上貫穿一個索引孔（INDEX HOLE）。

在切割後的第二個步驟是磁性媒體的打磨過程。通常這些被切成碟狀的磁性媒體在打磨之時，會被兩枚陶瓷製的光滑圓柱「碟」夾在中間，然後瓷碟作相反方向旋轉使達到打磨的目的。但這兩面瓷碟對磁性媒體打磨之時最重要的是控制壓力及接觸時間，使獲致最優質製成品——最高輸出及最低磁頭與磁性媒體接觸壓力。

完成打磨過程後，對優質磁性媒體而言便要作第二次舖設（POST COATING），這個過程是將一層很薄的潤滑劑複蓋在磁性媒體的表面之上。這種潤滑劑揮發性十分高，爲著要達到均勻的效果，會將磁性媒體旋轉浸入，令其兩個表面都被均勻複蓋，然後取出以高速旋轉，利用離心效應將多餘的潤滑劑分離，並藉此將表面的潤滑劑凝乾。

品質檢定

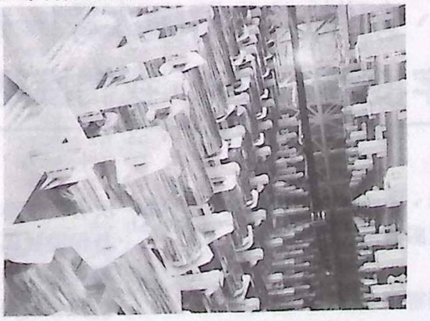
到目前爲止，磁性媒體的製作過程已告一個段落。這時磁性媒體會被放進磁碟外套並投入測試，作品質檢定。

對優質磁碟而言，每隻磁碟都需要單獨作一次全面性的質量檢定。方法是在磁碟的一切可用範圍內的每個位元位置寫入〝1〞，再經過其他儀器測試是否能將這些訊號全部讀回。假若在讀回訊號的過程中發現錯誤，製造商會將磁碟可能「故障」的部分重覆3次上述測試工序，通過後才將成功的產品投入下一個測試程序。

測試程序共分爲兩個主要部分：模擬特性測試及數字特



通常在踏入生產區之前，所有員工必須先經過一個「空氣浴室」將身上所沾染到的泥塵、微粒全部洗淨。

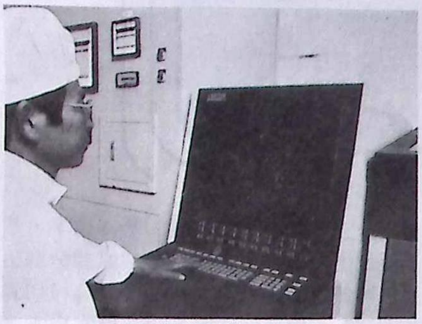


一張磁碟的質素好壞，採用的膠底非常重要，圖中所見的是未曾加上 COATING 的膠片。

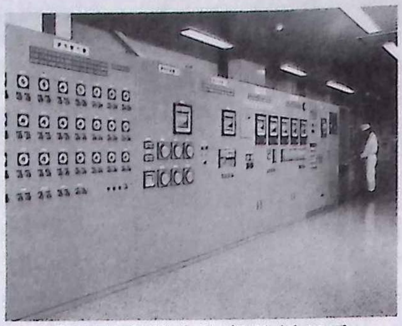


用來切割磁性物質的廠房必須是絕對無塵，否則生產出來的磁碟肯定不能通過品質試驗。

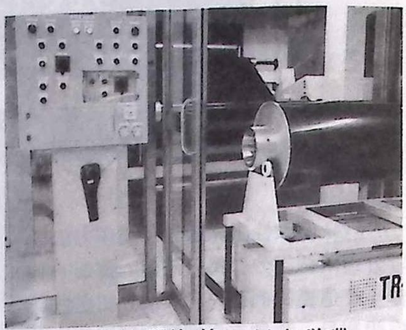
一張軟磁碟是怎樣誕生？



生產人員利用電腦來控制監察磁碟的品質。



任何磁碟的生產都是採用電腦化，如此方能得到品質的保證。



一卷卷的膠片將切割成磁碟的形狀，並準備加上磁性物質 COATING 。

性測試。現在讓我們先討論一下模擬特性測試的過程是如何進行的。

首先儀器會在磁性媒體的最外層磁軌（稱爲第0磁軌）上灌錄一系列測試訊號，然後再利用儀器將這些訊號讀回作三項測試，目的是檢定磁碟內磁性媒體的：

①響應性能：保證磁性媒體的輸出訊號與寫入訊號相同。

②輸出幅度：輸出幅度要足夠大才能確保電腦系統能明確地讀入磁性媒體上的訊號。

③脈冲幅度調制率：因爲優質的磁性媒體要求所有磁軌上的訊號輸出強度能夠平穩；換句話說，輸出強度變化是愈小愈好的。

當以上三項測試都在第0磁軌上圓滿通過後，測試器材便會對最內層磁軌作一次同樣的「質量檢定」。通常這條最內層磁軌即我們稱爲第39磁軌或第76磁軌（雙倍密度磁軌的磁碟機使用到）。這條最內層磁軌是質量檢定過程的焦點所在；因爲在這條磁軌上所灌錄的資料是最容易產生「不可靠」的。理由是這條磁軌的長度十分短，使資料貯存的密度相對提高，令到磁軌的上述三項模擬特性容易受到影响。

當磁碟成功通過上述測試後，便會進入數字特性測試程序。測試過程是利用儀器的讀／寫磁頭在磁性媒體上作不同程度的信號限幅測試。ECMA（EUROPEAN COMPUTER MANUFACTURERS ASSOCIATION）的最低限幅標準是35％。不過由於高貯存量的磁碟要求較嚴格的關係，製造商會由60％的限幅率（CLIPPING LEVEL）開始測試，然後將限幅率逐步調低，次序是50％，然後是40％，若果磁碟仍未能通過40％限幅率的測試，這一面的磁碟便會終止測試，然後重覆磁碟另一面的同樣測試，通過後便作爲單面使用（SINGLE SIDE）。

當然，假若磁性媒體在兩個表面上都能通過60％限幅率的測試程序，已可以確保無誤地用作雙面，雙倍密度（DOUBLE SIDE，DOUBLE DENSITY）的應用，例如IBM的8608規格——一種26磁區，512字節的雙面，雙倍密度灌錄格式。

檢定合格後的最後步驟是作一個稱爲「初始化」的處理（INITIALIZATION）。方法是在第0磁軌上灌錄一些訊號，使電腦系統可以鑑別這是最外層磁軌，然後在其餘磁軌上亦寫入一些訊號，經儀器再次核對無誤後，這枚磁碟便可送到用戶的手上了。■

要將個人的經驗、創作、心得讓大家一齊分享，惟一途徑就是將它化成文字投寄來本刊。我們接納任何形式的稿件。

投稿規則

㈠ 本刊歡迎任何形式的投稿，包括軟件、硬件評介，及製作，工商業應用、遊戲軟件評介，電腦應用技巧，電腦程序、語言編寫技巧。

㈡ 所有軟件程式投稿，必須要連同磁碟寄來或携來本出版社，程式必須附加清楚說明，用法及原理。交來之磁碟於文章發表後，即退回。如要退回原稿，先付回郵信封。

㈢ 稿件長度字數不限，著重言之有物。

㈣ 所有稿件必須附上作者中英文姓名，通訊電話及地址，身份證號碼，發表時筆名隨意。

㈤ 稿件必須用原稿紙橫排書寫，一字一格。

㈥ 請勿一稿兩投。

㈦ 稿酬於文章發表後即支持。

㈧ 來稿請寄九龍中央郵局信箱71193 號電腦時代收，信封請勿黏口（有磁碟者除外），信封面請註明〔投稿〕，或親身携來本出版社。

用左右箭咀鍵選擇的 CATALOG HELLO

時常看到不同的HELLO在磁碟中出現，但來來去去也總是以A、B、C、D…… 的形式出現，一些新意也沒有，偶然看到COPY][PLUS 的選擇功能，令筆者羨慕不已，心想如果有一個像這種選擇功能的HELLO 便好了，於是筆者便設計了這個INVERSE CATALOG。大家在BOOT 起磁碟後，只要利用左右箭咀便可以選擇執行需要的檔案。

用者只需用機械碼把程式輸入電腦，然後按入BSAVE INVERSE CATALOG，A$810，L$2CC，這樣便能輸入磁碟，而你亦可根據你的磁碟把這程式作適當的安排。

程式執行時，首先會顯示磁碟冊號、所用磁碟的空間、剩餘的空間，跟着隨即顯示檔案，用者只要把認爲重要的檔案鎖起來，便能在這時顯示，反之便不會顯示出來，還有用者如果把自己的名字連同程式INIT 磁碟的話，那麼當程式執行時，便會在INVERSE CATALOG的標題顯示出來。

值得一提的是筆者特別在這程式中加入FAST•LOAD（BLOAD）程式，如果用者把它連同DEVERSI DOS 或其他快DOS 使用，便能夠比原本的DEVERSI DOS 或其他的DOS 快了接近二秒，不信的話不妨試試。

相信讀者看完這篇文章後，便立刻把它輸入電腦並加以執行。

爲方便讀者將INVERSE CATALOG INIT 成一張碟，大家可使用附同的一個APPLESOFT 程式（見列表2，INIT INV. CATALOG）。

做法如下：

(1) 關掉電腦，再重新開啓電源。
(2) BOOT起SYSTEM MASTER。
(3) 打入NEW
(4) BLOAD入INVERSE CATALOG。
(5) LOAD入INIT INV. CATALOG。
(6) 打入INIT HELLO，按RETURN。
(7) 大功告成。

大家可以在今期同時出版的程式磁碟上找到本文的兩個檔案，名字分別是：

(1) INVERSE CATALOG
(2) INIT INV. CATALOG ■

列表 2 INIT 程式。

```
1  REM  ******************
2  REM  * INIT INV.HELLO *
3  REM  ******************
10 CALL 2064
```

列表 1 主程式：INVERSE CATALOG

```
 REM  ********************
 REM  ** INVERSE HELLO **
 REM  **  A$810,L$290  **
 REM  ********************
0810- 20 84 FE 20 2F FB 20 58
0818- FC A9 00 85 FF 85 F9 85
0820- FB 85 FD A9 20 85 FC A9
0828- 13 85 FA AD F7 B7 8D 19
0830- 0A AD F8 B7 8D 1A 0A A9
0838- 0F 8D 1D 0A A9 0A A0 18
0840- 20 D9 03 A9 0B 85 F9 A0
0848- 00 B1 F9 E6 F9 E6 F9 C9
0850- FF F0 33 A6 FF F0 09 B1
0858- F9 10 2B 29 07 9D 00 10
0860- E6 F9 B1 F9 91 FB C8 C0
0868- 1E D0 F7 A9 00 91 FB C6
0870- F9 A5 FC 9D 00 11 A5 FB
0878- 9D 00 12 18 69 20 85 FB
0880- D0 02 E6 FC E6 FF A5 F9
0888- 18 69 21 85 F9 D0 B8 CE
0890- 1D 0A D0 A8 A9 2D A0 0A
0898- 20 3A DB 20 CA 09 A2 05
08A0- 20 4A F9 8A A0 20 20 3A
08A8- DB 20 8E FD 20 CA 09 A9
08B0- 0A A0 18 20 D9 03 AC 06
08B8- 13 A2 09 20 F0 09 D0 F9
08C0- 20 D5 09 20 D5 09 E6 FD
08C8- E6 FD A5 FD C9 8C D0 F0
08D0- C6 FF D0 03 4C B3 09 C6
08D8- FF A2 00 86 FE A9 04 85
08E0- 22 85 25 20 FC 09 E6 25
08E8- A5 FE C5 FF F0 08 E6 FE
08F0- A5 25 C9 18 D0 ED A9 04
08F8- 85 25 A9 00 85 FE A0 20
0900- A9 08 20 A8 FC AD 30 C0
0908- 88 D0 F5 20 80 FE 20 FC
0910- 09 20 84 FE 2C 00 C0 10
0918- FB 20 FC 09 20 28 FD C9
0920- 95 F0 41 C9 88 F0 56 C9
0928- 8D D0 D3 20 BD 09 A9 56
0930- A0 0A 20 3A DB 20 03 0A
0938- 20 8E FD A9 84 20 ED FD
0940- A0 77 A6 FE BD 01 10 F0
0948- 07 A0 7C C9 04 F0 01 C8
0950- 98 A0 0A 20 3A DB A6 FE
0958- BD 01 12 BC 01 11 20 3A
0960- DB 4C 8E FD A5 FE C5 FF
0968- F0 94 E6 FE A5 25 C9 17
0970- D0 06 20 70 FC 4C 0B 09
0978- E6 25 4C 0B 09 A5 FE F0
0980- E7 C6 FE A5 25 C9 04 D0
0988- 25 A9 17 48 20 24 FC A5
0990- 28 85 2A A5 29 85 2B A0
0998- 27 68 E9 00 C9 03 F0 10
09A0- 18 48 20 24 FC B1 28 91
09A8- 2A 88 10 F9 30 E1 C6 25
09B0- 4C 0B 09 20 BD 09 A9 60
09B8- A0 0A 4C 3A DB 20 2F FB
09C0- 20 58 FC A9 0C 85 25 4C
09C8- 22 FC A2 28 A9 AD 20 ED
09D0- FD CA D0 F8 60 A0 08 A6
09D8- FD 1E 38 13 A2 1E B0 02
09E0- A2 27 20 F0 09 D0 F0 E6
09E8- FD 60 A9 B0 9D 00 04 CA
09F0- FE 00 04 BD 00 04 C9 BA
09F8- F0 F0 88 60 A9 00 85 24
0A00- 20 22 FC A6 FE BD 01 12
0A08- BC 01 11 20 3A DB A2 09
0A10- 20 4A F9 A0 27 91 28 60
0A18- 01 60 01 00 11 00 29 0A
0A20- 00 13 00 00 01 00 FE 60
0A28- 01 00 01 EF D8 D6 CF CC
0A30- D5 CD C5 BA B0 B0 B0 A0
0A38- A0 A0 A0 D3 C5 C3 D4 CF
0A40- D2 D3 A0 A0 C6 D2 C5 C5
0A48- BA B0 B0 B0 A0 D5 D3 C5
0A50- C4 BA B0 B0 B0 00 CC CF
0A58- C1 C4 C9 CE C7 A0 BA 00
0A60- 87 CE CF A0 D0 D2 CF C7
0A68- D2 C1 CD D3 A0 C1 D6 C1
0A70- C9 CC C1 C2 CC C5 00 C5
0A78- D8 C5 C3 00 C2 D2 D5 CE
0A80- 00 FF FF FF FF FF FF FF
0A88- FF FF FF FF FF FF FF FF
0A90- FF FF FF FF FF FF FF FF
0A98- FF FF FF FF FF FF FF FF
0AA0- 49
```

程式編寫技考入門之(一)

從六合彩到編寫程式技巧

— 蕭耀森 —

對於部份經驗較淺的APPLE 用戶來說，在編寫程序來解決問題時很可能會遇上不少困難。如果閣下自覺是以上的一類APPLE 用戶，筆者肯定這文章會對閣下有所帮助。在一般情况下，我們可以通過多參考別人的程序來增強本身的程序編寫能力。無疑，吸取別人所長，從而找出本身的缺點所在，再加以改進的學習途徑是千古不逆的。但是某些程序根基不穩的朋友們，可能會不知何者爲長，更甚者連程序也看不懂。因此筆者决定編寫一些有啓發性的程序，藉此灌輸程序編寫的知識。

在香港五百萬市民當中，相信沒有一位不被六合彩的風氣所感染。如果我們能編寫一個提供六合彩號碼的程序，必然會很有趣味。不若我們一同來編寫這程序吧！

從APPLE 用戶手册中，我們可以找到一個函數RND（RANDOM FUNCTION）。它可以隨意地給出一個大於或等於0，而小於1的實數（0≦RND(1)＜1）。然而我們需求的號碼是一個大於0及小於41的整數（1≦抽出號碼≦40），於是我們祇有把它乘以40，令它變爲一個大於等於0及小於40的實數，然後通過函數INT 來取小數點之前的整數部份，變爲一個介乎0與39的整數，加以1之後，便完全符合我們的要求了。由於投注六合彩是需要6個號碼，所以我們要引入一個以I爲變量的LOOPING 。至此，我們已有了以下的程序。

```
300  FOR I = 1 TO 6
400 NU =  INT (40 *  RND (1)) + 1
430  PRINT NU;"0":
600  NEXT I
730  PRINT
```

如果閣下認爲以上的程序已符合我們的要求，閣下便犯了一個錯誤。事實上，以上的程序是有BUG的，因爲它會給出一些相同的號碼。爲了除去這條臭蟲，我們必需把已抽出的號碼標下記號，並在每次抽出號碼後檢查它是否重覆。如果號碼已有標記（表示已抽出了）的話，便跳回第400行，再抽另一個號碼，直至抽出的號碼是第一次出現的，才可到達第430行，把它PRINT 出來。因此，我們要設置一個以DUP 爲名稱，並擁有40個元素的陣列（DIM DUP(40)）。跟着便把陣列內的元素設置爲0，表示相應的號碼還沒有出現。如果抽出的號碼NU是第一次出現時，我們便在PRINT出NU前把DUP（NU）設置爲1，表示這號碼已抽出了。加入了以上的改進後，我們的程序便變成

```
100  DIM DUP(40)
200  FOR I = 1 TO 40
210 DUP(I) = 0
220  NEXT I
300  FOR I = 1 TO 6
400 NU =  INT (40 *  RND (1)) + 1
410  IF DUP(NU) < > 0 THEN 400
420 DUP(NU) = 1
430  PRINT NU;"0";
600  NEXT I
730  PRINT
```

至此，我們的程序已相當圓滿。但人們總是要精益求精，不若我們再把程序增強，使到6個號碼依次序排列。在加進排序的部份前，我們的程序必然要作出或多或少的修改。首先是把第100行及430行改爲

```
100  DIM DUP(40),BOX(6)
430  BOX(I) = NU
```

並加入第700至720行來把號碼PRINT出。

```
700  FOR I = 1 TO 6
710  PRINT BOX(I);"0";
720  NEXT I
```

進行排序的方法有很多種，其中一個既直接又簡單的途徑是交换分類法（INTERCHANGE SORT）。但是這分類法的效率十分之低，而它亦爲人所熟悉，所以筆者設計了一種蛻變自BUBBLE 分類法的方法。

```
500 J = I
510  IF BOX(J - 1) < BOX(J) OR J =
     1 THEN 600
520 TEMP = BOX(J - 1)
530 BOX(J - 1) = BOX(J)
540 BOX(J) = TEMP
550 J = J - 1
560  GOTO 510
```

筆者的這個設計與一般的分類法有很明顯的分別，它可把動態（DYNAMIC）生成的元素排序，每當程序的上半部

從六合彩到編寫程式技巧

份抽出一個號碼後，便把這號碼與BOX 內的上一個號碼比較，若果次序正確的話，便表示BOX 內的元素是排列正確。不然的話，便把它們的位置對換，如此類推，直至對上的號碼是小於自己。除了這個分別之外，這個分類法還可以用數學歸納法來證明它的正確性。

到此爲止，我們所需要的一個提供六合彩號碼程式便算大功告成，大家不妨在學習之餘，利用本程式，選出六個幸運號碼！■

```
100  DIM DUP(40),BOX(6)
200  FOR I = 1 TO 40
210 DUP(I) = 0
220  NEXT I
300  FOR I = 1 TO 6
400 NU =  INT (40 *  RND (1)) + 1
410  IF DUP(NU) <  > 0 THEN 400
420 DUP(NU) = 1
430 BOX(I) = NU
500  J = I
510  IF BOX(J - 1) < BOX(J) OR J =
    1 THEN 600
520 TEMP = BOX(J - 1)
530 BOX(J - 1) = BOX(J)
540 BOX(J) = TEMP
550 J = J - 1
560  GOTO 510
600  NEXT I
700  FOR I = 1 TO 6
710  PRINT BOX(I);" ";
720  NEXT I
730  PRINT
```

捉蟲訓練——

又一個列印質數的程式

以下是一個簡單的程式，用來列出由二至用家輸入數字之間的所有質數；但它在執行時，卻只能顯示出十以內的質數，究竟岔子出在那裏呢？

```
10  INPUT N
20  HOME : HTAB 16: PRINT "PRIME
    NO.": PRINT
30  PRINT 2" "3" ";
40  FOR I = 3 TO N STEP 2
50  FOR J = 3 TO  SQR (I)
60  IF I / J =  INT (I / J) THEN
    90
70  NEXT
80  PRINT I" ";
90  NEXT
100  PRINT : PRINT : PRINT "THAT'
    S ALL !!": END
```

我們歡迎讀者提供〔捉蟲訓練〕問題。程式長度沒有限制，以越簡短越好。程式必須要以APPLESOFT 寫成，並符合下列各項要求：

1. 程式必須是APPLESOFT ，任何POKE或CALL機械語言的敍述必須正確。
2. 程式必須能出現某些結果，而不可出現SYNTAX ERROR。
3. 所有問題必須附加簡短說明及答案。

假如閣下的作品獲得刊登，我們將會送出十二期免費雜誌給你或將你的訂閱延長一年。

所有投稿請寄九龍中央郵政信箱C. P.O BOX 71193 。信封面請寫上〔投稿捉蟲〕。■

答案：

只要將第九十行的NEXT 改爲 NEXT I便DEBUG成功了。錯處驟眼不易發覺，因爲習慣上NEXT 後面不加變數名；但在此處，當 第行 60爲眞時，跳到行 90，APPLE 誤以爲行 90的NEXT 爲 NEXT J，BUG便撤了出來——還是懶之過！■

圖像/文字顯示選擇器

邵英傑

APPLE 電腦超卓的圖像顯示，的確吸引了不少電腦愛好者，在坊間筆者亦見不少關於電腦繪圖的書籍，可謂各式其式，花多眼亂，但筆者至今觀看了多部此類書籍後，雖然不乏精彩之作，但發覺它們大多數都沒有詳細的介紹——「如何選擇適當的圖像顯示模式」。當然，在MANUAL 內是有提及，但祇是一個與其相關的表列，且偏隅一角，這對於初學者來說，根本不能明白，甚至是學機器語言或組合語言的人，有時也會無所適從，故此筆者寫了一個簡淺的組合語言程式（PROGRAM 1），希望對各位有所帮助。

這個程式，主要的功用是帮助我們選擇畫面，例如：選擇低解像第一頁顯示；又或是選擇高解像第二頁加文字幕顯示，APPLE 的10種圖像顯示模式，均可由此程式運作達成。程式有三個版本，分別是BASIC語言、機器語言和LISA組合語言，大家可根據所需加以選用。

程式亦盡量簡化選擇畫面的程序，若使用機器語言或組合語言，以開啓高解像全畫面的第一頁爲例，必須先後觸動位址＄C050、＄C057、＄C054、＄C052，使用的畫面多了，則需要觸動由位址＄C050至＄C057多次，其實分別記憶各位址的用途，也是不容易。現在我們祇需將不同的值（由＄０至＄Ｆ）放入Ａ暫存器，然後呼叫SET GRAPHIC便成，它自會作出反應，例如值＃$08選擇高解像全畫面第一頁，各位可根據表列一作出選擇。

若是使用APPET SOFT BASIC 的用家，可根據程式二，將行號10至行號20載入程式，因它們是用來設定「&」指標及將SET GRAPHIC 程式衍入記憶體內，如此我們便可以「&８」選擇高解像第一頁，又或是以「&E」選擇高解像第二頁加文字幕顯示。不過使用「&」之後不可留有空格，而且「&」之後祇容許一個指令碼，選擇數由０至F，請參看表列一。

對於BASIC 用家，此程式還有另外一些優點，因BA—SIC中選擇圖像模式是用GR或HGR，但它們都會先清除畫面，因此以前利用PLOT或HPLOT指令畫成的圖樣，便不能保留，而且祇容許有五種圖像選擇，分別是TEXT，GR，GR2，HGR和HGR2，對於有十種顯示模式的APP—LE 電腦，實在不敷應用，若使用此SET GRAPHIC 程式，便可解決以上的困難。

列表一

HEX	圖像顯示
0	低解像、第一頁
2	低解像、第一頁、加文字幕
4	低解像、第二頁
6	低解像、第二頁、加文字幕
8	高解像、第一頁
A	高解像、第一頁、加文字幕
C	高解像、第二頁
E	高解像、第二頁、加文字幕
1，3 9，B	全文字幕、第一頁
5，7 D，F	全文字幕、第二頁

程式一爲組合語言

```
!ASM
**END OF PASS 1

**END OF PASS 2

0300                    ORG $300
0300            **********************************
0300            *                                *
0300            *         SET GRAPHIC            *
0300            *         -----------            *
0300            *       By Shao Ying Kit         *
0300            *                                *
0300            **********************************
0300            *                                *
0300            *  ENTER: Use A-register as      *
0300            *           instructor           *
0300            *                                *
0300            **********************************
0300 29 0F      SET.GR    AND #$F
0302 85 2B                STA $2B
0304 A9 00                LDA #$0
0306 85 2A                STA $2A
0308 A2 00                LDX #$0
030A A5 2B      ONCEMORE  LDA $2B
030C 86 3E                STX $3E
030E E0 00                CPX #$0
0310 F0 04                BEQ CONTINUE
0312 6A         ROTATE    ROR
0313 CA                   DEX
0314 D0 FC                BNE ROTATE
0316 A6 3E      CONTINUE  LDX $3E
0318 29 01                AND #$1
031A 05 2A                ORA $2A
031C A8                   TAY
031D B9 50 C0             LDA $C050,Y
0320 E6 2A                INC $2A
0322 E6 2A                INC $2A
0324 E8                   INX
0325 E0 04                CPX #$4
0327 D0 E1                BNE ONCEMORE
0329 60                   RTS
032A A0 00      GO.ON     LDY #$00
032C B1 B8                LDA ($B8),Y
032E 49 30                EOR #$30
0330 C9 0A                CMP #$0A
0332 90 07                BCC DIGIT
0334 38                   SEC
0335 E9 67                SBC #$67
0337 C9 10                CMP #$10
0339 B0 03                BCS RETURN
033B 20 00 03   DIGIT     JSR SET.GR
033E E6 B8      RETURN    INC $B8
0340 D0 02                BNE DONE
0342 E6 B9                INC $B9
0344 60         DONE      RTS
0345                      END

***** END OF ASSEMBLY
```

像圖顯示器

```
 1  REM  ************************
 2  REM  * SET GRAPHIC (BASIC) *
 3  REM  ************************
10  DATA  41,15,133,43,169,0,133
    ,42,162,0,165,43,134,62,224,
    0,240,4,106,202,208,252,166,
    62,41,1,5,42,168,185,80,192,
    230,42,230,42,232,224,4,208,
    225,96,160,0,177,184,73,48,2
    01,10,144,7,56,233,103,201,1
    6,176,3,32,0,3
15  DATA 230,184,208,2,230,185,96
20  FOR I = 768 TO 836: READ A: POKE
    I,A: NEXT : POKE 1013,76: POKE
    1014,42: POKE 1015,3
25  REM  HGR1 + TEXT
30  & A
35  REM  HGR2
40  FOR I = 1 TO 1000: NEXT
50  & C
60  FOR I = 1 TO 1000: NEXT
65  REM  TEXT1
70  & 8
```

程式二為BASIC語言

```
***********************
* SET GRAGHIC (M.C) *
*    A$300, L$345     *
***********************
0300- 29 0F 85 2B A9 00 85 2A
0308- A2 00 A5 2B 86 3E E0 00
0310- F0 04 6A CA D0 FC A6 3E
0318- 29 01 05 2A A8 B9 50 C0
0320- E6 2A E6 2A E8 E0 04 D0
0328- E1 60 A0 00 B1 B8 49 30
0330- C9 0A 90 07 38 E9 67 C9
0338- 10 B0 03 20 00 03 E6 B8
0340- D0 02 E6 B9 60 00
```

程式三為機器語言

■

用RAM咭來選擇遊戲的LEVEL

LAM KIN MING

參閱第十七期的"WILD CARD與RAM 咭的配合應用"一文後，雖然有幾個GAME 都可以修改LEVEL 或人數而任你玩，但都要利用WILD CARD，在參閱同期的"有保護遊戲之拆圖方法"後，就靈機一觸，只是利用RAM咭就能將兩個GAME 拆解，KARATE—KA和TAPPER。因為很多電腦使用者都沒有WILD 咭而只有RAM咭，所以這個方法十分方便。

KARATEKA：

①先放一張普通有FILE 的碟放入機中BOOT機
②CALL—151
③C081 C081
④D000<D000.FFFFM
⑤C083 C083
⑥FFFC：00 03
⑦C080
⑧放入KARATEKA磁碟
⑨C600G

當畫面出現第一個LEVEL 後，如你想立刻玩下一個LEVEL 就按下CTRL 鍵和RESET 鍵，之後就可以跳進下一個 LEVEL，如是者，每次按CTRL和.RESET 鍵就可跳去下一個LEVEL。但要注意，必須在畫面出後才可按鍵。

TAPPER：

按着前面第①至第⑤個步驟做
⑥FFFC：59 FF
⑦C080
⑧放入TAPPER磁碟
⑨C600 G

等你選擇好LEVEL和JOYSTICK後，等待至開始玩時，就按CTRL 和RESET 鍵。之後由第⑩個步驟繼續做。

⑩1E76：7E 7E 7E
⑪D00G

之後遊戲會繼續，而你會有128 次機會，最後祝你好運。■

SLOT 0 以外的16 K RAM咭用途

從十七期刊登的「有保護程式的拆圖方法」，自己心裏深深不安，雖然用SLOT 0上的RAM CARD 可拆很多圖，但始終對於一些「猛」GAME 無用武之地，試想，一個程式BOOT 起後，它可以隨時用C081，C081 把RAM咭上的資料改變，但想深一層，RAM CARD是插在SLOT 0上，如果RAM CARD插在其他SLOT上，則不易被那「猛」GAME所悉破，雖然把RAM CARD 插在其他SLOT ，人人都識，但怎樣去開啓却是一個問題，但皇天不負有心人，終於在一本書上，查到

SLOT 1：C081→C091
C083→C093

SLOT 2：C081→C0A1
C083→C0A3

只要依照上面的方法更改，你將會擁有更多美麗而光榮的圖像，在此暫別，GOODBYE！■

ROM-MAN

書信文件編打器

電腦最大的功能之一是資料處理，其中文字處理器對於辦公室自動化貢獻良多。一般的文字處理器如WORDSTAR均需要學習一段時間方可熟識使用，但假如將這些處理器掉開一旁不用，可能很快便會忘記了它的操控方法。對於只是間中需要編打文件的朋友來說，這些處理器並不好受用。

爲了方便這些並非經常要編打文件的朋友，筆者編寫了一個簡單易用，絕對不用死背操控指令而又功能強效的文字處理器——文件編打處理器。有了這個文件編打器後，閣下的打字機可拋掉一旁了！利用各個控制鍵，閣下能隨意清除畫面上所有東西、隨意在螢幕上移動游標，設定反白、閃爍顯示，當然你可好像使用IBM 電動打字機一樣，預先設定TABULATION。

在執行文件編打處理器時，每次可最多打入8頁的資料內容。閣下可隨意翻前、翻後各頁 、倒換兩頁 、加一新頁在兩頁之中或清除該頁。所有這些文件全部可自由存入磁碟或從磁碟上取回使用。程式內置有一個文件名INDEX 供檢索各項的文件資料。假如閣下擁有一部印字機，即可隨時將這些文件頁編印出。

執行本程式後，會出現一空白螢幕，右上角處顯示有PAGE 1。當鍵入字母時，螢幕會印出字母。當游標走到右邊緣時，它會再從新在左邊出現，如要跳到下一行，需要按CTRL－M或RETURN 。當打到螢幕底部最尾時，電腦不會像往常一般，上捲一行供你繼續打下去，取而代之是游標會再次在螢幕左上角出現。閣下如欲開新一頁，需要按CT－RL P。以下是本文所有的控制鍵（所有控制鍵需先按〔CTRL〕）

A：將游標向上移
B：將游標移到底線
C：清除該頁
D：從記憶中刪除一頁
E：〔留空將來使用〕
F：檔案存取
G：〔留空將來使用〕
H：將游標左移
I：加插一頁
J：翻看前一頁
K：下一頁
L：〔留空將來使用〕
M：即RETURN
N：將所有記憶中內容清洗
O：將游標移原處（即左上角）
P：開新一頁
Q：脫離程式
R：重置水平TABULATION
S：印出印字機
T：水平TABULATION（1—9）
U：游標右移
V：選擇顯示模式
W：觀看指令控制鍵總表
X：調換頁數
Y：〔留待未來使用〕
Z：游標下移

水平TABULATION

T 1 ＝5字位	T 6 ＝30字位
T 2 ＝10字位	T 7 ＝35字位
T 3 ＝15字位	T 8 ＝40字位
T 4 ＝20字位	T 9 ＝1個字位
T 5 ＝25字位	

本程式只有一個FILE：大家打入後可用SAVE PAGE EDITOR 存入磁碟。■

```
1  REM  ************************
2  REM  *      PAGE.EDITOR     *
3  REM  * BY   COMPUHOUSE      *
4  REM  *                      *
5  REM  * COPYRIGHT (C) 1985 *
6  REM  *                      *
7  REM  ************************
110  HOME
120  VTAB 10: HTAB 16: PRINT "PAG
     E EDITOR"
140  HIMEM: 28928: POKE 32,0: PRINT
     : VTAB 22: PRINT "PRESS ANY
     KEY TO CONTINUE: ";: GET A$:
      PRINT : GOTO 1950
150  VTAB (V): HTAB (H): GET I$:I
      =  ASC (I$): IF I > 31 THEN
     180
160  ON I GOTO 220,250,270,290,35
     0,370,720,740,770,810,840,87
     0,890,920,950,970,1000,1030,
     1090,1220,1250,1280,1370,139
     0,1430,1450
170  CALL 64484: GOTO 150
180  IF V = 24 AND H = 40 THEN I =
      PEEK (2038): HTAB (39): PRINT
     I$;: POKE 2039, PEEK (2038):
      POKE 2038,I: GOTO 200
190  PRINT I$;
200 H = H + 1: IF H > 40 THEN H =
     1:V = V + 1: IF V > 24 THEN
     V = 1
```

```
210  GOTO 150
220  REM  A=UP
230 V = V - 1: IF V < 1 THEN V =
    24
240  GOTO 150
250  REM  B=BOTTOM
260 H = 1:V = 24: GOTO 150
270  REM  C=CLEAR
280  GOSUB 1930: GOTO 150
290  REM  D=DELETE CURRENT PAGE
300  IF MP = 1 THEN  CALL 64484: GOTO
     150
310  GOSUB 1480: HOME : VTAB (10)
     : PRINT "DELETE THIS PAGE, R
     EALLY (Y/N)?";: GET I$: PRINT
     I$: GOSUB 1520: IF I$ < > "
     Y" THEN 150
320 I = CP - 1:MP = MP - 1: IF I =
     0 THEN I = 1
330  IF CP = 1 + MP THEN CP = I: GOSUB
     1520: GOTO 150
340 CP = CP + 1: GOSUB 1480:CP =
     CP - 1: GOSUB 1480:CP = CP +
     1: GOTO 330
350  REM  E=<FUTURE USE>??
360  GOTO 170
370  REM  F=FILES
380  GOSUB 1480: HOME : VTAB (3):
     PRINT "FILES ACCES OPTIONS"
     : PRINT
390  PRINT "1 LOAD PAGE EDITOR FI
     LE": PRINT "2 SAVE PAGE EDIT
     OR FILE": PRINT "3 DELETE PA
     GE EDITOR FILE": PRINT "4 SO
     RT PAGE EDITOR FILE NAMES": PRINT
     "5 RETURN TO ENTRY SCREEN": PRINT
     : PRINT "OPTION?";: GET I$: PRINT
     I$
400 I =  VAL (I$): IF I < 1 OR I >
     4 THEN  GOSUB 1520: GOTO 150
410  ON I GOTO 420,500,600,670
420  REM  LOAD FILE
430 F = 1: GOSUB 1630:I =  VAL (I
     $): IF I < 1 OR I > N THEN  PRINT
     "YOU MUST ENTER A VALID NUMB
     ER": FOR I = 1 TO 1000: NEXT
     : GOSUB 1520: GOTO 150
440  ONERR  GOTO 470
450  PRINT D$;"BLOAD"; MID$ (N$(I
     ),2, LEN (N$(I)));".PE"
460  POKE 216,0:MP =  VAL (N$(I))
     :CP = 1: GOSUB 1520: GOTO 15
     0
470  POKE 216,0: IF  PEEK (222) <
     > 6 THEN  PRINT "ERROR NUMB
     ER "; PEEK (222): PRINT "PLE
     ASE CONSULT YOUR MANUAL": FOR
     I = 1 TO 2000: NEXT : GOSUB
     1520: GOTO 150
480  PRINT "THAT NAME IS NOT ON T
     HIS DISK": PRINT "SHOULD I D
     ELETE THAT ENTRY (Y/N)?";: GET
     I$: PRINT I$: IF I$ <  > "Y"
     THEN  GOSUB 1520: GOTO 150
490  FOR II = I TO N - 1:N$(I) =
     N$(I + 1): NEXT :N = N - 1: GOSUB
     1580: GOSUB 1520: GOTO 150
500  REM  SAVE PAGE EDITOR FILE
510 F = 0: GOSUB 1630:I =  VAL (I
     $): IF I <  > 0 THEN I$ =  MID$
     (N$(I),2, LEN (N$(I)))
520  FOR X = 1 TO N: IF I$ <  >  MID$
     (N$(X),2, LEN (N$(X))) THEN
     NEXT :N = N + 1:X = N
530 N$(X) =  STR$ (MP) + I$:X = N
540  ONERR  GOTO 570
550  PRINT D$;"BSAVE";I$;".PE,A$7
     600,L";1024 * MP: GOSUB 1580
560  GOSUB 1520: GOTO 150
570  POKE 216,0: PRINT : PRINT : IF
     PEEK (222) = 11 THEN  PRINT
     "ILLEGAL FILE NAME": GOTO 59
     0
580  PRINT "ERROR NUMBER "; PEEK
     (222): PRINT "PLEASE CONSULT
     YOUR MANUAL"
590 N = N - 1: FOR I = 1 TO 2000:
     NEXT : GOTO 560
600  REM  DELETE FILE
610 F = 1: GOSUB 1630:I =  VAL (I
     $): IF I < 1 OR I > N THEN  PRINT
     "YOU MUST ENTER A VALID NUMB
     ER": FOR I = 1 TO 1000: NEXT
     : GOSUB 1520: GOTO 150
620  ONERR  GOTO 650
630  PRINT D$;"DELETE"; MID$ (N$(
     I),2, LEN (N$(I)));".PE": GOSUB
     1580
640  POKE 216,0: FOR II = I TO N -
     1:N$(I) = N$(I + 1): NEXT :N
     = N - 1: GOSUB 1520: GOTO 1
     50
650  POKE 216,0: IF  PEEK (222) <
     > 6 THEN  PRINT "ERROR NUMB
     ER "; PEEK (222): PRINT "PLE
     ASE CONSULT YOUR MANUAL": FOR
     I = 1 TO 2000: NEXT : GOSUB
     1520: GOTO 150
660  GOTO 640
670  REM  SORT PAGE EDITOR NAMES
680  FOR I = 1 TO N:SL = I: FOR I
     I = I TO N
690  IF  MID$ (N$(SL),2, LEN (N$(
     SL))) >  MID$ (N$(II),2, LEN
     (N$(II))) THEN SL = II
700  NEXT :H$ = N$(SL):N$(SL) = N
     $(I):N$(I) = H$: NEXT
710  GOSUB 1580: GOSUB 1520: GOTO
     150
720  REM  G=<FUTURE USE>
730  GOTO 170
740  REM  H=MOVE LEFT (LEFT ARROW
     )
```

```
750 H = H - 1: IF H < 1 THEN H =
    40
760  GOTO 150
770  REM  I=INSERT NEW PAGE
780  IF MP = 8 THEN  CALL 64484: GOTO
    150
790  GOSUB 1480:II = CP: FOR I =
    MP TO II STEP  - 1:CP = I: GOSUB
    1520:CP = I + 1: GOSUB 1480:
     NEXT
800 MP = MP + 1:CP = II: GOSUB 19
    30: GOTO 150
810  REM  J=BACKWARD PAGE
820  GOSUB 1480:CP = CP - 1: IF C
    P < 1 THEN CP = MP
830  GOSUB 1520: GOTO 150
840  REM  K=FORWARD PAGE
850  GOSUB 1480:CP = CP + 1: IF C
    P > MP THEN CP = 1
860  GOSUB 1520: GOTO 150
870  REM  L=<FUTURE USE>
880  GOTO 170
890  REM  M=RETURN
900 H = 1:V = V + 1: IF V > 24 THEN
    V = 1
910  GOTO 150
920  REM  N=NEW
930  GOSUB 1480: HOME : VTAB (4):
     PRINT "NEW, REALLY (Y/N)?";
    : GET I$: PRINT I$: IF I$ <
    > "Y" THEN  GOSUB 1520: GOTO
    150
940 MP = 1:CP = 1:VV = 0:H = 1:V =
    1: GOSUB 1930: GOTO 150
950  REM  O=ORIGIN
960 H = 1:V = 1: GOTO 150
970  REM  P=PAGE
980  IF MP = 8 THEN  CALL 64484: GOTO
    150
990  GOSUB 1480:MP = MP + 1:CP =
    MP: GOSUB 1930: GOTO 150
1000  REM  Q=QUIT
1010  GOSUB 1480: HOME : VTAB (4)
    : PRINT "QUIT, REALLY (Y/N)?
    ";: GET I$: PRINT I$: IF I$ <
     > "Y" THEN  GOSUB 1520: GOTO
     150
1020  GOSUB 1580: END
1030  REM  R=RESET HORIZ. TABS
1040  GOSUB 1480: HOME : VTAB (4)
    : PRINT "HORIZONTAL TABULATI
    ONS": VTAB (8): FOR I = 1 TO
    9: PRINT "T";I;" = "; RIGHT$
    ("0" +  STR$ (T(I)),2): NEXT
    : VTAB (21): HTAB (1): PRINT
    "ENTER NEW VALUE OR 'RETURN'
    "
1050  VTAB (8): FOR I = 1 TO 9: HTAB
    (6): GET IW$: IF  ASC (IW$) =
    13 THEN 1070
1060  PRINT IW$;: GET I2$:T(I) =
     VAL (IW$ + I2$): IF T(I) <
    1 OR T(I) > 40 THEN T(I) = 1
1070  HTAB (6): PRINT  RIGHT$ ("0
    " +  STR$ (T(I)),2): NEXT
1080  GOSUB 1520: GOTO 150
1090  REM  S=SCREEN DUMP
1100  GOSUB 1480: VTAB (1): PRINT
    : PRINT D$;"PR#1":I = 1
1110  FOR IW = 1024 TO 1104 STEP
    40
1120  FOR I2 = IW TO IW + 896 STEP
    128
1130  VTAB (I):I = I + 1: FOR I3 =
    I2 TO I2 + 39:C =  PEEK (I3)
1140  IF C > 127 THEN 1190
1150  IF C > 95 THEN C = C + 64: GOTO
    1190
1160  IF C > 63 THEN C = C + 128:
    GOTO 1190
1170  IF C > 31 THEN C = C + 128:
    GOTO 1190
1180 C = C + 192
1190  PRINT  CHR$ (C);
1200  NEXT : PRINT : NEXT : NEXT
1210  PRINT D$;"PR#0": GOSUB 1520
    : GOTO 150
1220  REM  T=HORIZ. TAB
1230  GET I$: IF I$ < "1" OR I$ >
    "9" THEN  CALL 64484: GOTO 1
    50
1240 H = T( VAL (I$)): GOTO 150
1250  REM  U=MOVE RIGHT (RIGHT AR
    ROW)
1260 H = H + 1: IF H > 40 THEN H =
    1
1270  GOTO 150
1280  REM  V=VIDEO
1290 VV = VV + 1: IF VV > 2 THEN
    VV = 0
1300  GOSUB 1310: GOTO 150
1310  ON VV + 1 GOTO 1320,1330,13
    40
1320  NORMAL : GOTO 1350
1330  INVERSE : GOTO 1350
1340  FLASH
1350 VF =  PEEK (1062): IF VF = 1
    60 OR VF = 96 OR VF = 32 THEN
     VTAB 1: HTAB 39: PRINT " ";
1360  RETURN
1370  REM  W=WHAT?
1380  GOSUB 1480: GOSUB 1760: GOSUB
    1520: GOTO 150
1390  REM  X=EXCHANGE CP WITH CP+
    1
1400 I = CP:CP = 0: GOSUB 1480:CP
     = I + 1: IF CP > MP THEN CP
    = 1
1410  GOSUB 1520:CP = I: GOSUB 14
    80:CP = 0: GOSUB 1520:CP = I
     + 1: IF CP > MP THEN CP = 1
1420  GOSUB 1480:CP = I: GOSUB 15
    20: GOTO 150
```

```
1430  REM  Y=<FUTURE USE>
1440  GOTO 170
1450  REM  Z=MOVE DOWN
1460 V = V + 1: IF V > 24 THEN V =
     1
1470  GOTO 150
1480  REM  STORE CURRENT PAGE
1490 T1 =  PEEK (50):T2 =  PEEK (
     243): NORMAL
1500 IW = 4 * (CP - 1) + 118:I2 =
     4 * (CP - 1) + 121
1510  POKE 60,0: POKE 61,4: POKE
     62,255: POKE 63,7: POKE 64,2
     55: POKE 65,I2: POKE 66,0: POKE
     67,IW: CALL 768: RETURN
1520  REM   RESTORE CURRENT PAGE
1530  POKE 50,T1: POKE 243,T2
1540 IW = 4 * (CP - 1) + 118:I2 =
     4 * (CP - 1) + 121
1550  POKE 60,0: POKE 61,IW: POKE
     62,255: POKE 63,I2: POKE 64,
     255: POKE 65,7: POKE 66,0: POKE
     67,4: CALL 768
1560  IF PEEK (1058) = 208 AND PEEK
     (1060) = 199 AND  PEEK (1062
     ) = 160 THEN  POKE 1063,176 +
     CP
1570  RETURN
1580  REM  SAVE INDEX
1590  PRINT D$;"OPEN INDEX.PE"
1600  PRINT D$;"WRITE INDEX.PE"
1610  PRINT N: FOR I = 1 TO N: PRINT
     N$(I): NEXT
1620  PRINT D$;"CLOSE": RETURN
1630  REM  PICK DISK NAME
1640  HOME : IF N = 0 AND F = 1 THEN
     VTAB (10): PRINT "NO NAMES
     PRESENT": FOR I = 1 TO 1000:
     NEXT : GOSUB 1520: POP : GOTO
     150
1650  IF N <  > 0 THEN 1680
1660  VTAB (20): PRINT "ENTER NAM
     E TO STORE";: INPUT I$: IF I
     $ = "" THEN  GOSUB 1520: POP
     : GOTO 150
1670  RETURN
1680 I$ = "": FOR I = 1 TO N: PRINT
     RIGHT$ ("  " + STR$ (I),3)
      + "   " +  MID$ (N$(I),2, LEN
     (N$(I))): IF  INT (I / 20) =
     I / 20 THEN  GOSUB 1720
1690  NEXT : IF I$ = "" THEN  GOSUB
     1720
1700  IF I$ = "" THEN  GOSUB 1520
     : POP : GOTO 150
1710  RETURN
1720  PRINT : PRINT "ENTER THE NU
     MBER ";: IF F = 0 THEN  PRINT
     "OR NAME ";
1730  PRINT "YOU WANT": PRINT "OR
     'RETURN' ONLY";: INPUT I$: IF
     I$ <  > "" THEN I = N
1740  IF  VAL (I$) < 0 OR VAL (I
     $) > N THEN  PRINT "ENTER A
     NUMBER BETWEEN 1 AND ";N: GOTO
     1720
1750  RETURN
1760  REM  HELP
1770  GOSUB 1480: HOME : VTAB (3)
     : PRINT "NIBBLE PAGE EDITOR
     CONTROL KEYS": PRINT
1780  PRINT "A=MOVE CURSOR UP    N
     =NEW"
1790  PRINT "B=CURSOR BOTTOM     O
     =CURSOR TO ORIGIN"
1800  PRINT "C=CLEAR PAGE        P
     =CREATE NEW PAGE"
1810  PRINT "D=DELETE PAGE       Q
     =QUIT"
1820  PRINT "E=<FUTURE USE>      R
     =RESET HORIZ. TABS"
1830  PRINT "F=FILES ACCESS      S
     =SCREEN DUMP"
1840  PRINT "G=<FUTURE USE>      T
     =HORIZ. TAB (1-9)"
1850  PRINT "H=CURSOR LEFT       U
     =CURSOR RIGHT"
1860  PRINT "I=INSERT A PAGE     V
     =VIDEO SWITCH"
1870  PRINT "J=PREVIOUS PAGE     W
     =WHAT? (THIS LIST)"
1880  PRINT "K=NEXT PAGE         X
     =EXCHANGE PAGES"
1890  PRINT "L=<FUTURE USE>      Y
     =<FUTURE USE>"
1900  PRINT "M=RETURN            Z
     =CURSOR DOWN"
1910  PRINT : PRINT "HORIZONTAL T
     ABULATIONS": PRINT : FOR I =
     1 TO 5: PRINT "T";I;"= "; RIGHT$
     ("0" +  STR$ (T(I)),2);" ";:
      NEXT : PRINT : FOR I = 6 TO
     9: PRINT "T";I;"= "; RIGHT$
     ("0" +  STR$ (T(I)),2);" ";:
     NEXT
1920  PRINT : PRINT : PRINT "PRES
     S ANY KEY?";: GET I$: PRINT
     I$: HOME : RETURN
1930  REM  HOME AND DO PAGE
1940  HOME : HTAB (35): NORMAL : PRINT
     "PAGE ";CP: GOSUB 1310: RETURN
1950  REM  INIT ALL
1960 V = 1:H = 1:CP = 1:MP = 1:D$
     =  CHR$ (4): DIM N$(100)
1970  FOR I = 1 TO 8:T(I) = I * 5
     : NEXT :T(9) = 1
1980  ONERR  GOTO 2020
1990  PRINT D$;"OPEN INDEX.PE"
2000  PRINT D$;"READ INDEX.PE"
2010  INPUT N: FOR I = 1 TO N: INPUT
     N$(I): NEXT
2020  PRINT D$;"CLOSE": POKE 216,
     0
2030  POKE 768,160: POKE 769,0: POKE
     770,76: POKE 771,44: POKE 77
     2,254: REM     SET UP MOVE WI
     TH SHORT MACHINE CODE
2040  GOSUB 1760: GOSUB 1930
2050  GOTO 150
```

配合PRINT SHOP使用

印製賀咭信封套

APPLE GREETING CARDS

德理

筆者相信許多讀者都擁有PRINT—SHOP這個軟件，而且會感到非常實用。PRINTSHOP內置的功能之中包括有印製各種各樣的賀咭，可惜唯一美中不足的是它只帮你印製賀咭，但却沒有設計可即時印出裝載賀咭的信封套。雖然閣下可以使用其他現成的信封套，但由於用PRINTSHOP 印出來的賀咭在呎吋上並不與外面坊間出售的信封套成標準，不是過長，便是太濶，總是衣不稱身。爲了這原故，筆者特别編寫出這個印製信封套的程式，好讓製出來一個GOOD FIT 呎吋的信封。

這個程式特别之處不單可以輕易印出一個有間線的未摺前信封套，而且更可以在編印前，輸入收件人姓名及地址（共有四行文字輸入）。程式特色之一是這幾行文字輸入，會自動較中間位左右對稱，換言之印出來的效果非常美觀。程式之所以有此功能，全靠行句1010。各位不妨研究。

編者按：我們徵求讀者修改此程式，令到有下列功能：

(1) 將整張印字紙9吋半摺成爲一個信封，中間信封面可隨意輸入收件人地址及名稱。

(2) 可先輸入多個人名及地址，然後逐一印出。該等人名及地址可收錄在磁碟並隨時再用回。

(3) 在左上角或左下角可印出寄信人的回郵地址或公司名稱。

(4) 印出之收件人名字及地址可印出大字或DOUBLE WIDTH。■

```
10  GOSUB 790
20  REM   ** PRINT SHOP ENVELOPE
30  REM   ** MAJOR PROGRAM SECTIO
    NS
40  REM   - 20, TEXT DISPLAY
50  REM   - 50, CENTER STRING
60  REM   - 55, MAKE FRAME
70  REM   - 60, WAIT FOR KEYPRESS
80  REM   - 70, PRINT SUBS.
90  REM   - 90, EXIT SUB.
100 REM   - 100, MAIN PROGRAM
110 REM   - 200, EXIT OPTIONS
120 GOSUB 250:NAME$ = "PRINT SHO
    P ENVELOPE MAKER": VTAB 3: GOSUB
    240
130 NAME$ = "--------------------
    -----": GOSUB 240
140 PRINT : HTAB 3: PRINT "THE G
    REETING CARDS PRODUCED BY PR
    INT"
150 HTAB 3: PRINT "SHOP (BRODERB
    UND) DON'T FIT REGULAR"
160 HTAB 3: PRINT "ENVELOPES.  T
    HIS PROGRAM WILL PRINT"
170 HTAB 3: PRINT "A PATTERN THA
    T YOU CAN CUT AND FOLD,"
180 HTAB 3: PRINT "THEN GLUE OR
    TAPE INTO AN ENVELOPE.": PRINT
    : PRINT : PRINT
190 NAME$ = "TURN ON YOUR PRINTER
    ": GOSUB 240:NAME$ = "(START
    AT THE TOP OF A SHEET)": GOSUB
    240: PRINT
200 NAME$ = "PRESS SPACE BAR TO C
    ONTINUE,": GOSUB 240: PRINT
210 NAME$ = "OR ESCAPE KEY FOR EX
    IT OPTIONS": GOSUB 240
220 GOSUB 260: IF A$ =  CHR$ (27
    ) THEN  GOTO 650
225 IF KP = 32 THEN  LET FLAG =
    1
230 PRINT : PRINT : PRINT : PRINT
    : GOTO 430
240 CEN = (20 -  INT (( LEN (NAME
    S$)) / 2)): HTAB CEN: PRINT
    NAME$: RETURN
250 HOME : VTAB 1: INVERSE : PRINT
    SPC( 40): VTAB 23: PRINT  SPC(
    40): FOR I = 1 TO 23: VTAB I
    : HTAB 1: PRINT  SPC( 1): NEXT
    : FOR I = 23 TO 1 STEP - 1:
    VTAB I: HTAB 40: PRINT  SPC(
    1): NEXT : NORMAL : RETURN :
    REM   FRAME
260 KP =  PEEK (49152): IF KP < 1
    28 THEN 260
270 KP = KP - 128:A$ =  CHR$ (KP)
    : POKE 49168,0: RETURN
```

封套的製造

```
280  PRINT  SPC( 8);: FOR I = 1 TO
     64: PRINT ".";: NEXT : PRINT
     : RETURN : REM   LINE 1, 19,
     & 47
290  PRINT  SPC( 8);"."; SPC( 62)
     ;".": RETURN : REM   LINES 2
     - 18 & 48 - 59
300  PRINT  SPC( 4);D3S$; SPC( 62
     );D3S$: RETURN : REM   LINE
     20 & 46
310  PRINT D7S$; SPC( 62);D7S$: RETURN
     : REM   LINES 21 - 41
320  PRINT D7S$; SPC( DIS);MESS$;
     SPC( 63 - (DIS +  LEN (MESS
     $)));D7S$: RETURN : REM   LI
     NES 29-32
330  PRINT D7S$; SPC( 42);LN$ + D
     7S$: RETURN : REM   LINES 42
     - 45
340  PRINT  SPC( 12);"."; SPC( 16
     );AP$; SPC( 16);".": RETURN
     : REM   LINE 60
350  PRINT  SPC( 16);: FOR I = 1 TO
     48: PRINT ".";: NEXT : PRINT
     : RETURN : REM   LINE 61
360 D$ =  CHR$ (13) +  CHR$ (4)
370  ONERR  GOTO 390
380  HOME : PRINT D$"RUN MENU"
390  ONERR  GOTO 410
400  PRINT D$"RUN HELLO"
410  PRINT D$"CATALOG"
420  POKE 216,0: VTAB 23: END
430 D3S$ = ".   .": REM   DOTS &
     3 SPC'S
440 D7S$ = ".       .": REM   DOT
     'S & 7 SPC'S
450 AP$ = ">APPLE GREETING CARDS<
     ": REM   22 CHAR.
460 LN$ = "____________________":
     REM   20 U/L'S
470  PR# 1: GOSUB 280: REM   LINE
     1
480  FOR N = 1 TO 17: GOSUB 290: NEXT
     : REM   LINES 2 - 18
490  GOSUB 280: REM   LN 19
500  GOSUB 300: REM   LINE 20
510  FOR N = 1 TO 8: GOSUB 310: NEXT
     : REM   LINES 21 - 28
515  IF FLAG = 1 THEN  GOTO 1000
520 MESS$ = M$(1): GOSUB 780: GOSUB
     320: REM   LINE 29
530 MESS$ = M$(2): GOSUB 780: GOSUB
     320: REM   LINE 30
540 MESS$ = M$(3): GOSUB 780: GOSUB
     320: REM   LINE 31
550 MESS$ = M$(4): GOSUB 780: GOSUB
     320: REM   LINE 32
560  FOR N = 13 TO 21: GOSUB 310:
     REM LINE 33 - 41
570  FOR N =  1 TO 4: GOSUB 330: NEXT
     : REM   LINES 42 - 45
580  GOSUB 300: REM   LN 46
590  GOSUB 280: REM   LN 47
600  FOR N = 1 TO 12: GOSUB 290: NEXT
     : REM   LN 48 - 59
610  GOSUB 340: REM   LN 60
620  GOSUB 350: REM   LN 61
630  PRINT  CHR$ (12);: REM   FOR
     MFEED
640  PR# 0: CALL 1002
650  GOSUB 250
660  VTAB 10: HTAB 12: PRINT "1 -
     RUN AGAIN"
670  VTAB 12: HTAB 12: PRINT "2 -
     ADD MESSAGE TO ENVELOPE"
680  VTAB 14: HTAB 12: PRINT "3 -
     EXIT TO BASIC"
690  PRINT : PRINT :NAME$ = "SELE
     CT BY NUMBER": GOSUB 240
700  GOSUB 260:N =  VAL (A$):FLAG
     =  VAL (A$)
710  IF N < 1 OR N > 3 THEN  PRINT
     CHR$ (7);: GOTO 700
720  ON N GOTO 120,730,420
730  GOSUB 250: VTAB 10: HTAB 5: PRINT
     "ENTER YOUR FIRST LINE MESSA
     GE": VTAB 11: HTAB 5: INPUT
     M$: IF ( INT ( LEN (M$) / 2)
     * 2) =  LEN (M$) THEN  GOSUB
     775
735 M$(1) = M$
740  VTAB 12: HTAB 5: PRINT "ENTE
     R YOUR SECOND LINE MESSAGE":
     VTAB 13: HTAB 5: INPUT M$: IF
     ( INT ( LEN (M$) / 2) * 2) =
     LEN (M$) THEN  GOSUB 776
745 M$(2) = M$
750  VTAB 14: HTAB 5: PRINT "ENTE
     R YOUR THIRT LINE MESSAGE": VTAB
     15: HTAB 5: INPUT M$: IF ( INT
     ( LEN (M$) / 2) * 2) =  LEN
     (M$) THEN  GOSUB 777
755 M$(3) = M$
760  VTAB 16: HTAB 5: PRINT "ENTE
     R YOUR FORTH LINE MESSAGE": VTAB
     17: HTAB 5: INPUT M$: IF ( INT
     ( LEN (M$) / 2) * 2) =  LEN
     (M$) THEN  GOSUB 778
765 M$(4) = M$
770  GOTO 120
774  REM   LINE 775-778 TO PREVEN
     T EVEN NO
775 M$ = M$ + " ": RETURN
776 M$ = M$ + " ": RETURN
777 M$ = M$ + " ": RETURN
778 M$ = M$ + " ": RETURN
780 DIS = (32 - ( INT ( LEN (MESS
     $)) / 2)): RETURN
790  REM   TITLE PAGE
800  TEXT : HOME : HTAB 9: INVERSE
     : PRINT "
     ": NORMAL : HTAB 9: INVERSE
     : PRINT " PRINT SHOP ENVELOP
     E ": NORMAL : HTAB 9: INVERSE
     : PRINT "
     ": NORMAL
810  VTAB 5: HTAB 18: PRINT "BY":
     VTAB 7: HTAB 14: PRINT "ALB
     ERT NG": VTAB 9: HTAB 12: PRINT
     "COPYRIGHT 1985"
820  VTAB 23: HTAB 5: PRINT "PRES
     S <SPACE BAR> TO CONTINUE.";
     : GET K$: PRINT : RETURN
1000  FOR N = 1 TO 4: GOSUB 310: NEXT
     : GOTO 560
```

小型投資家適用

股票 投資分析系統



如果你在股票市場上是個小投資家的話，則STOCK ANALYST　程式必定是你夢寐以求的輔助工具。它評估你的股票、讓你清楚知道自己的投資概況，並助你回答何時應買入？何時賣出？何時停止投資？等等之類的問題。另外，此程式在使用上是非常簡易的，用家毋須具備專業電腦知識。

程式會在STOCK'S HISTORY 中爲每一種股票提供一個資料檔案。此HISTORY 是儲存有關股票的交易資料，而當股票被買入或賣出時，所有的資料是會自動UPDATED 的。而STOCK ANALYST　主程式就是利用此資料去籌備製出三個報告表。

此三個報告表分別是股票狀況報告表（STOCK STATUS REPORT）、股票歷史報告書（STOCK HISTORY REPORT）及股票證券報告表（STOCK PORTFOLIO）。STATUS REPORT是利用最UPDATED　的資料去計算股票的賺或虧。而HISTORY REPORT 是列出股票自買入以來的交易情況、資料。最後，PORTFOLIO REPORT是將每隻股票的STATUS　REPORT合併起來，再計算出其總投資是賺錢或虧本。

需要設備

STOCK ANALYST 程式是必須在具有48K或以上記憶的蘋果電腦，至少有一部磁碟機，並能在DOS 3.3或PR－ODOS系統下運行。另外，你更可以利用印字機印出報告表，但這並不是必須的。

程式用法詳析

當你RUN這個程式時，圖一的選擇表（MENU）會出現。在選擇表上會有五個選擇，在此你可以利用左右箭咀來選擇，而〔RETURN〕鍵是在當你選擇後按的。這種選擇方

```
MAIN MENU:

     UPDATE A STOCK'S STATUS
     REPORT MENU
     INITIALIZE A STOCK
     ENTER DISK COMMANDS
     QUIT

     DISK DRIVE........................... 1
     DISK SLOT............................ 6
     PRINTER SLOT......................... 1
     RAM PRINTER DRIVER............... NO
     OUTPUT DEVICE............... SCREEN

ARROWS MOVE  'RETURN' TO SELECT/MODIFY
```

```
REPORT MENU:

     RETURN TO MAIN MENU
     INDIVIDUAL STOCK STATUS
     INDIVIDUAL STOCK HISTORY
     PORTFOLIO STATUS
     QUIT

     DISK DRIVE........................... 1
     DISK SLOT............................ 6
     PRINTER SLOT......................... 1
     RAM PRINTER DRIVER............... NO
     OUTPUT DEVICE............... SCREEN

ARROWS MOVE  'RETURN' TO SELECT/MODIFY
```

法毋須要你輸入數字，又省回不少時間！

在表中的五個選擇號，用家可以利用左右箭咀及RETU－RN 鍵去更改這些參數。

輸入資料應注意事項

1.當程式要求你輸入數字（指價格）的資料時，用家必須利用標準的模式（例如："PRICE PER SHARE？" 12.5）。切記！你不需要輸入錢的符號（$）！

2.只有在 "PRICE PER SHARE？" 問題下，你才可輸入包括小數點或分數的數字。用分數時必須使用下列形式 12½，要將分數½及整數12分開一格，而用 "／" 符號來分開分母及分子。

3.此程式在顯示數字方面會使用正常的形式（例如：＄1，198，40），故用家在輸入時不須輸入＄，。…等符號，程式會自行加上。

程式選擇表之用法：

A）開始儲存一個STOCK之資料：

顧名思義，選擇這個程序是輸入一個在檔案中沒有的STOCK 資料。首先，你會被要求輸入公司名，股票持有人的代表符號或簡稱及股票的交易地方。由於你所輸入的公司名是會被作爲檔案名稱的（FILENAME），故你所輸入的名稱不能超過十二個字。接着，程式會要求你輸入 "初買股票的詳細資料，例如：購買日期、股票數量、每股之價值、經紀公司的佣金等及其他有關之資料。最後，當檔案建立後，你會回到主選擇表。

B. UPDATE 股票的STATUS：

此選擇容許你UPDATE 你所擁有股票的數量及價值。在你RUN一個股票的REPORT之前，你會想UPDATE 這個程式的STATUS。首先利用CURSOR BAR 從INDEX 中

由於程式列表過長，這裏並不刊出本程式列表，大家可於今期出版的程式磁碟中找到本程式。

選擇你要UPDATE 的股票。而在股票名稱之後的是最後UPDATE 的日期。在你輸入今天日期及股票價值後，程式會詢問你在今日是否有進行任何有關此股票的交易；在這裏，交易是指增購股票、賣出股票、派息等等。之後這些新資料會被加在檔案上，並且你會再次回到主選擇表。

另外，有一點要補充的是，此功能是容許你從PORTFOLIO中刪去一股票檔案；但請注意，一旦一個股票檔案被刪去，你是沒法再使它RECOVER的！

C. 使用磁碟指令：

此選擇使你能在此程式下使用磁碟指令。例如，如有須要你可在此模式下BLOAD一個PRINTER'S DRIVER 的ROUTINE。你又可以CATALOG，VERIFY、RENAME、UNLOCK一個檔案等。而你只須按〔RETURN〕便能回到主選擇表了。

D. REPORT選擇表

不用多說，選擇此功能使你去到REPORT 選擇表。（參考圖二）

REPORT 選擇表的用法

接着我們談談REPORT選擇表的用法。

A. 個別的股票STATUS：

此選擇提供你在PORTFOLIO 中個別股票的STATUS資料。例如股票的"背景資料"（BACKGROUND DATA）及最UPDATE 的STATUS 資料。在表中，"INVESTMENT TO DATE" 欄中是投資者爲了這股票而使用的一切總支出；"RETURN TO DATE" 是賣出股票而得到的總數；"NET INVESTMENT TO DATE" 就是上述兩者之差額；"VALUE OF HOLDING" 則是計出現時股票的總值（股票數量×每股之現時售價）；而最後便是"PROFIT／LOSS了，它將"NET INVESTMENT TO DATE" 與"VALUE OF HOLDING" 比較而計算出此項投資在現時來講是賺或虧。而在RUN完此 REPORT後，你會回到REPORT 選擇表。

B. 個別股票之歷史（HISTORY）：

個別股票之歷史表列出某指定股票之所有交易資料；而交易資料是分爲三類：i）買入更多股票ii）賣出股票及 iii）股票派息。若果股票是因爲派息或買入而增加的話，則此交易資料會被放在第一類別欄下。同樣的，在RUN過此HISTORY，你也會回到REPORT 選擇表。

C. 有價證券類（PORTFOLIO）之STATUS：

此表給你股票有價證券類的一個價值觀念。此表也是唯一的一個表在螢幕顯示，與印字機顯示是不同的；由於螢幕是受制於一行40字（除非使用80－COLUMN CARD）的，故在印字機輸出是比較詳細的。此表是將所有的個別股票STATUS 一次印出；已告訴你有價證券的價值、價格、賺或虧的數目等資料。之後，你也會回到REPORT 選擇表。

D. 回到主程式選擇表：

顧名思義，選擇此功能會令你回到主程式選擇表（MAIN MENU）。■

實例：
大家可從圖3至圖5看到一個實例：
圖(3)是一間名爲COMPUHOUSE 公司的股票買賣資料。
圖(4)是COMPUHOUSE 這隻股票的買賣歷史。
圖(5)則是利用印字機列印出來的一份PORTFOLIO。

```
            ** STATUS REPORT **

COMPUHOUSE (COM)
EXCHANGE: FAR EAST STOCK EXCHANGE LTD (FESEL)

INITIAL PURCHASE ON 05/01/85
100 SHARES AT $100 PER SHARE
ADDITIONAL FEES/CHARGES: $50.00
-----------------------------------------
STATUS AS OF 05/15/85:
-----------------------------------------
NUMBER OF SHARES OWNED: 110
PRICE PER SHARE: $95

INVESTMENT TO DATE            $19,600.00
RETURN TO DATE                 $9,170.00
NET INVESTMENT TO DATE      + $10,430.00

VALUE OF HOLDING              $10,450.00
PROFIT                          + $20.00
```

```
STOCK: COMPUHOUSE (COM)
EXCHANGE: FAR EAST STOCK EXCHANGE LTD
=========================================
 ACQUISITIONS
DATE      #SHARES  PPS          AMOUNT
05/01/85  100      100       10,000.00

    AGENT'S COMMISSION           50.00
    ODD LOT COST                  0.00
    SEC. EXCHANGE FEE             0.00
    TOTAL COST               10,050.00

05/15/85  100      95         9,500.00

    AGENT'S COMMISSION           50.00
    ODD LOT COST                  0.00
    SEC. EXCHANGE FEE             0.00
    TOTAL COST                9,550.00

-----------------------------------------
 SALES
DATE      #SHARES  PPS       NET REC'D
05/06/85  90       103        9,170.00
```

```
                                                     PREPARED 05/15/85

     PORTFOLIO STATUS REPORT

                                     PRICE    VALUE OF    NET INV.
     COMPANY       ABBREV  EXCH # SHR  PR SHR  HOLDING     TO DATE     PROF/LOSS
     ----------------------------------------------------------------------------

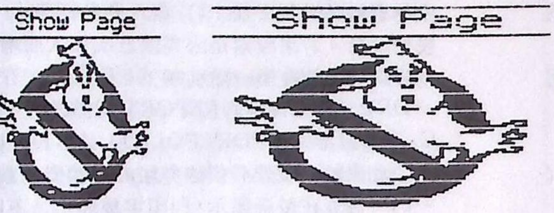
     COMPUHOUSE    COM     FESEL110    95      10,450.00   10,430.00   +20.00


     TOTALS:                                   10,450.00   10,430.00   +20.00
```

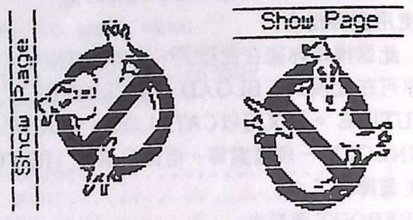
BEAGLE BROS 又一榮譽貢獻：

印字機魔術師：TRIPLE DUMP

胡永達



利用MAGNIFY X及Y 指令製造出來的圖形效果，其中圖形濶度放大了兩倍。



利用ROTATE 指令控制繪印出不同角度的圖形效果。

各位讀者，你有沒有發覺市面上支持印字機之軟件程序甚少，導致閣下在使用印字機時，會感到各方面有所不便或者難以徹底發揮印字機優處。由於印字機的性能有異，而且機上之控制指令繁多，並沒有較大之統一，讀者每每要花上一段時日才能熟習一部印字機全部控制指令。

基於這種種不便之因素，TRIPLE－DUMP 就在市面上應運而生。這個極爲實用之程序是BEAGLE BROS 又一榮譽新產品。筆者以第一時間爲讀者介紹這一個實用程序之用法及其優點。

TRIPLE－DUMP 適用於DOS 3.3及PRODOS系統下。程序主要分爲兩部份，(1)主程序－TRIPLE－DUMP;(2)附帶程序－BIGBANNER（海報大字）。首先BOOT起TR－IPLE－DUMP ，繼而按入〝R〞開始使用程序，然後選擇所需要之功能（在後加以詳述）及載入所需要印出之圖畫或文字。在此之前閣下可選擇印字機型號，印字咭類別及－SLOT 位置。按動電腦的印出掣就可以將你心目中喜愛的圖像或文字編印出來。

本程序之主要鍵掣有：

← →－移動浮標到閣下所需要使用之功能或所需要更改其數據之位置上。

SPACE BAR－觀看剛才載入之圖像，但首先要選擇了圖形是在高解像或是在低解像頁之中。

F 鍵－給印字機命令將紙張移上一頁。

L 鍵－給印字機命令將紙張移上一行。

ESC －離開主程序或是從副程式返回主程序。

各項强大功能

TRIPLE－DUMP最大的優點是程序內之功能，以下逐一詳述：

CHANGE PRINTER－因印字機之不同，控制指令就有很多差別，這一項功能內置有十多種印字機型號，大家只要選擇正確的印字機型號，便可將圖像或文字編印出來而毋需要鍵入繁多之印字機指令。換言之，只要携帶着TRIPLE－DUMP，就算在學校之中，朋友家中……使用陌生的印字機也不再成爲問題。

CHANGE INTERFACE－這項功能是更改印字咭，讓程序適用於閣下之印字機。程序內已經載入了數十款之印字咭，必有一款適合你。

PRINT SLOT －這項功能是選擇其印字咭之位置，就算印字咭不是插在SLOT 1 也不是問題。在這項功能內，可以利用RETURN 來選擇其印字機之位置。如果閣下之電腦是APPLE ⅡC ，印字咭之位置必須改爲SLOT 1。

PAPER WIDTH －這項功能是去選擇紙張之濶度。一般是由四吋至十六吋半，利用RETURN 鍵來增加紙張之濶度，每次增加半吋單位。當然紙張濶度絕對不能大過其印字機之濶度。

DELAY －這項功能是讓印字機印頭每當印完一行後，停止一段時間，防止印字頭過熱之用，通常0.5 秒則已經足夠了。但是當印出圖畫時，發覺每行有一針特別深色時，則應該增加DELAY時間至一秒或秒半也無妨。

LINEFEED －這項功能是命令印字機的印頭每當印完一行之後，返回起點才印第二行因爲一些比較低廉之印字機在使用穿梭式編印圖形時，往往會發現在每行之間留有

一條的罅隙，即俗稱「漏針」。雙重之LINEFEED 會使圖形在每行之間印多一行空白位。

SAVE PRINTER INFO 一以上之各項功能數據，可以利用這項功能一一的輸入磁碟上，留待日後再需要用到TRI—PLE—DUMP 時使用。

TYPE—選擇印出之圖像或文字模式，共有六項可供選擇，但必須要匹配電腦機種。六項模式如下：

(1) HI—RES
(2) LO—RES
(3) 40字行TEXT
(4) DOUBLE HI—RES
(5) DOUBLE LO—RES
(6) 80字行TEXT

其中第4、5及6項只適用於Ⅱe及Ⅱc。

LOAD PICTURE 一將磁碟上之檔案載入記憶。本副程序不能載入已壓縮過之圖形FILE 。大家可在選擇時，只須按動浮標到適當位置，然後按RETURN 即可。

CROP PICTURE—如果讀者們只想把圖形之某部份印出，往往要花上很多功夫才能做到，但是擁有這程序，則是輕而易擧之事。在顯示器中，可以看到四邊都有一條閃動之直線，只要將這四條直線移動到圖形某部份之四邊便可。控制這四條閃動的直線鍵掣如下：

L 一代表左方之閃動直線
R 一代表右方之閃動直線
T 一代表上方之閃動直線
B 一代表下方之閃動直線
〔←〕，〔→〕一將閃動之直線左、右、上、下移動。
〔,〕，〔.〕一將閃動之直線快速移動
ESC—返回原本之圖形大小

PRINT PICTURE 一將浮標移到此項而按下RETURN 時，程序就會自動根據TYPE之選擇，把記憶中之圖形或文字印出來。在印出圖像或文字之前，可按ESC取消。

INDENT 一在這項功能中輸入數據，就可決定會使圖形或文字在印出時，左邊MARGIN 線之濶度選擇範圍由0—9999。高解像以點（PIXEL）作單位，低解像以塊（BLOCK） 作單位，而文字就以字元作單位。

ROTATE 一在這項功能中，只有高解像圖形才能適用，首先把浮標移到這功能上，按RETURN 鍵就可以把高解像圖形作四種不同的方向旋轉，分別爲0度，90度，180度及270度。

NEGATIVE— 在這項功能中，只有高解像圖形才能適用，用RETURN鍵來選擇YES或NO。YES是代表將圖形以反白印出，而NO是代表將原本之圖形印出。

MAGNIFY X—在這項功能中，只有高解像圖形才能適用，只要輸入數據，圖形就可以很容易地把X軸放大，範圍由1—255。

MAGNIFY Y—這項功能與上項一樣，但只是將Y軸放大，範圍也是由1—255。

DENSITY—在這項功能中，只有高解像之圖形才能適用，按下RETURN 去選擇圖深淺顏色，高數值是代表比較深色，而低數值代表比較淺色。

DISK DRIVE 一按下RETURN去選擇使用DRIVE 1 或DRIVE 2 。

附帶一提，近期BEAGLE BROS 的產品大都加入了一個新指令—&，這個指令是命令電腦跳進指定之二進制地址。在TRIPLE—DUMP之程序內，浮標之移動及聲响是靠：& BEEP，& SCRUP，& SCRDOWN，& KEYBO—ARD和& REVERSE。讀者們不妨發掘“&”之奧妙。

附送程式：海報大字

在TRIPLE—DUMP 附帶着另外一個程序—BIGBA—NNER。顧名思義是印出很大之標語字句。BIGBANNER是可以適用於DOS 3.3及PRODOS中。首先BOOT 起此程序，然後輸入要印出之句語文字，繼而按下RETURN 就可以印出來。所有大字是由@組成，每個字母大小爲一頁印字紙（即8吋左右）。本程式不能印出細楷。讀者們如果不想用@符號來構成字母，可以改動程序內行句1010行號之@符號，用其它例如：A，B …等代替。

除了BIGBANNER外，TRIPLE—DUMP 所附帶之程式還有PICTURE PACKER 。這不但可以將圖形檔案壓縮以節省磁碟之貯存量，並可以將已經壓縮之圖形還原。其用法可見下例實例：

實例

```
        TO PACK A PICTURE
 10 HGR:PRINT CHR$(4);"BLOAD PICTURE,
    A$2000" (load your picture)
 20 V=64 (specifices 16384 as the storage
    location:64*256=16384)
 30 PRINT CHR$(4);"BLOAD PACK.HIRES"
    (unless it's already loaded)
 40 POKE 0,0:POKE 1,V
 50 CALL 24576 (packs the picture)
 60 LNGHT=PEEK(0)+PEEK(1)*256-V*256
    (packed lenght)
 80 PRINT CHR$(4);"BSAVE PICTURE,PACA"
    V*256;",L"LNGTH

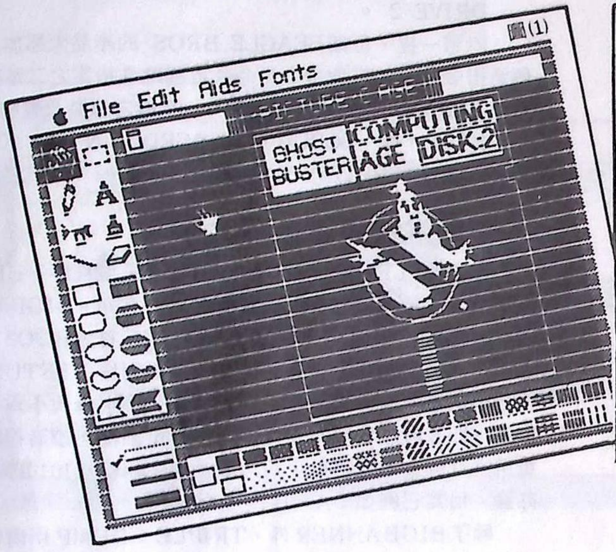
    Alternate values used for 64

        TO  UNPACK A PICTURE
 10 V=64 (specifies 16384 as the storage
      location:64*256=16384)
 20 HGR:PRINT CHR$(4);"BLOAD PICTURE,
    PACA";V*256
 30 PRINT CHR$(4);BLOAD PACK HIRES"
    (unless it's already loaded)
 40 POKE 0,0:POKE1,V (says unpack from
    page 2)
 50 CALL 24579 (unpacks the picture)

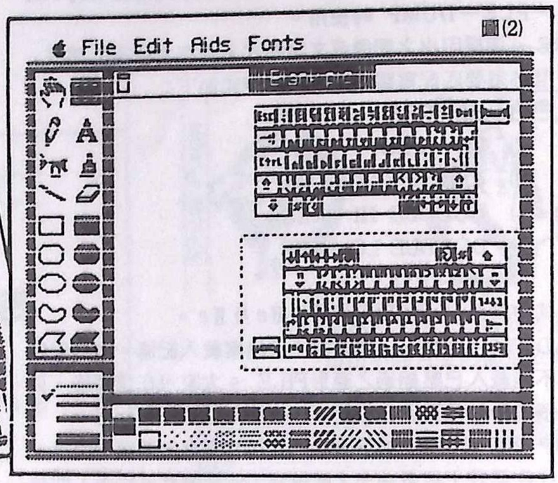
    Alternate values may be used for 64
 Zero-page location 230,set by HGR or
 HGR2,determines where the unpacked
 picture will be located.
```

穿梭蘋果的APPLE MOUSE 和MOUSE PAINT

中國人



圖(1)



圖(2)

如果閣下使用過LISA或MACINTOSH等新一代電腦，都會被它們的方便易用和高速所吸引。但我們必須面對事實，在APPLE Ⅱ上我們已放下了不少的投資，包括時間和金錢，所以不可能隨便地放棄它而選擇另一部祇試用了一會兒的電腦。話說過來，老鼠仔這傢伙在LISA或MAC 上實在扮演了一個極重要的角色，因它令一切的操作和使用都變得簡單。

有見及此，蘋果公司便於去年年底推出了APPLE Ⅱ的APPLE MOUSE。這使我們也能分享到LISA 的新技術。APPLE Ⅱ版本的售價是$150 美元（ⅡC版本只需$100美元），算得上是物超所值了。APPLE ⅡC 的用戶會發覺他們的版本售價比其他的平五十美元，但運行速度亦較其他型號緩慢，原因是：用於其他APPLE 型號的介面卡，它包括有一個第二微處理器，使到運行的速度加快。

除了老鼠仔的硬件部份外，還附帶一套名爲MOUSE PAINT 的多變化繪圖程式。假如你曾經見過MAC PAINT 程式在MACINTOSH 上的示範，那麽，筆者可以告訴大家，MOUSE PAINT 的功能和官感享受，正相等於MAC PAINT 的百份之七十八，對於一部加速了的APPLE Ⅱ 它在運行MOUSE PAINT 時，其敏捷程度簡直與MAC PAINT在MAC 上的情形不相伯仲。

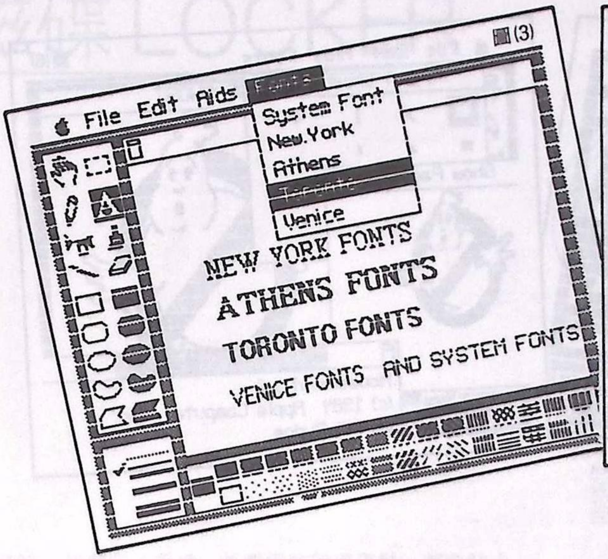
MOUSE PAINT 是一個運行PRODOS 的程式，可以在軟磁碟系統，或是全新的PRO FILE 硬碟機上運行自如。它使用了一個以ICON 爲主體的操作系統（ICON－ORIENT－ED OPERATING SYSTEM)，與及標準化的拉下菜單（PULL－DOWN－MENU)，還有可重叠的「視窗」等。

在啓動系統前，我們必須將介面卡插上，且必須在電腦旁邊騰出一平方呎的地方給它，它才能夠揮洒自如。「老鼠仔」的結構，是一個小盒子，騎在一個包着塑膠的滾軸球上，並有一個嵌在塑膠外殼上的大按鈕，當然，還有一條小不了的「尾巴」——一條與電腦相連的棕色電線。

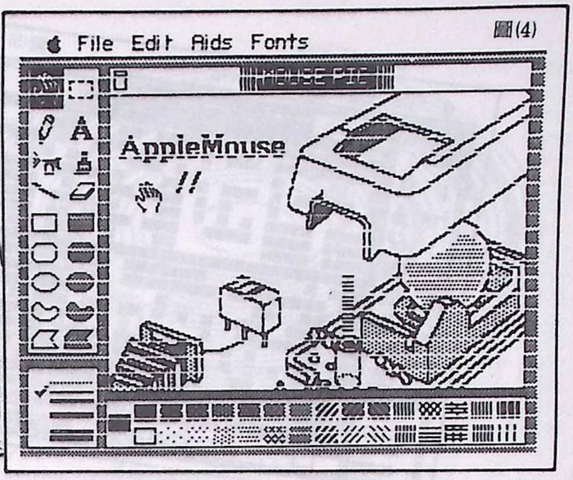
當你開啓系統的時候，你首先得到一個教授或程式本身的選擇權。假如你是選擇教授程式，它便會教你使用「老鼠仔」的方法。教授程式同時會教你如何使用拉下菜單（每一個菜單都有一個盒子載着的字，顯示在螢幕上端）。假設你將箭咀移到多項選擇菜單的盒子上，然後按下「老鼠仔」，這個菜單便會像拉下一幅窗簾一樣，被拉下來了。假如你想從菜單中選出"FATBITS"，那就必須繼續移動「老鼠仔」，在菜單上進行搜索，當你越過一個選擇時，螢幕的背景便會轉暗。記着，按掣一經鬆手後，選擇便會立即執行。假如你想回復原狀，只需將箭咀移離選擇項目，並鬆脫按掣便成。

經過簡短的課程後，你便可以開始進入程式，從螢幕上，你可看到的包括有：横排在螢幕頂部的五個拉下菜單（蘋果，FILE，EDIT，AIDS，FONTS）；位於左下方的十八種繪圖工具及線條選擇表；與及三十種排列在底部的背景圖案。至於"色板"則圍繞在繪畫範圍的四週，可即時顯示眞實影像的四份之三。

繪圖工具包括有：手（HAND）（見圖1），用來顯示更多的繪圖紙。鉛筆（PENCIL），讓你在螢幕上用手（指透過移動老鼠仔）來畫圖。噴霧（SPRAY），用來改變圖畫的光暗度。畫筆（BRUSH），用來畫出一些黑色的粗／幼



圖(3)



圖(4)

線。直邊（STRAIGHT EDGE），在任何位置上畫出長度不同的直線，可以直、橫及斜的。如果你不滿意畫出的圖案，膠擦（ERASER）可以爲你擦去圖畫的任何一部份。

另外的五組圖案，分別是：長方形，圓角長方形，橢圓／圓形，多邊形和自由形狀（FREE FORM）。你可以單畫它們的空心外形線條或是一個實心的形狀。

編輯箱（EDITOR BOX），可以在螢幕上移來移去，是一個具有多種尺碼的閃動長方形，可將要修改的部份圈起，再拉下EDIT菜單，利用CUT－PASTE和COPY命令，去將BOX內的複製品覆蓋在螢光幕上；或是利用INVERT命令將他們顚倒；或是利用其他EDIT的命令，如FLIP HORIZONTALLY 將他們左右倒轉或FLIP VERTICAL－LY將他們上下倒轉（見圖2）。

文字（TEXT），是由一個大的A字代表。你可以拉下頂部的FONTS拉下菜單去選擇五種不同的字體，再將指針放到適當的地方，這時所有鍵盤能提供的字都會被接受，包括Ⅱe的DELETE在內。和MAC PAINT不同，MOUSE PAINT 所提供的字體是沒有形式（STYLES）或尺碼（SIZES）的調校。這五種字體包括：標準型（SYSTEM FONT）、紐約（NEW YORK）、雅典（ATHENS）、多倫多（TORONTO）和威尼斯（VENICE）的字體（見圖3）。

在你正式開始繪畫之前，你首先要在FILE 菜單中選擇清洗螢幕（BLANK HIRES SCREEN）或是取回圖畫（GET PICTURE），因爲在開始時，是沒有繪畫紙的。MO－USE PAINT有一個完整的圖畫，就是那個你在說明書封面上所見過的「老鼠仔」硬件完全展示圖表，名稱是〝MO－USE PIC〞（見圖4）。

接着下來，閣下便可利用以上所說的十八種繪圖工具，色板，和字體等來繪出你所喜歡的圖畫。若果你需要作精密細緻的繪圖時，可以使用FATBITS 命令去擴大螢幕上的某一指定部份（見圖5），進行逐點修改，與此同時，螢幕的左上方會出現原圖的正常大小，方便你作爲參考。記着，除了SHOW PAGE命令外，其餘的都可以在FATBITS 命令上進行，如CUT、PASTE、COPY 等。

MOUSE PAINT 還有一個特色，就是在一個圖形下面的舊有東西全部保留下來，祇要選擇UNDO 這個命令，就會把新繪的圖形擦去，並恢復原有的舊圖形。但 UNDO 祇對最後的一個命令有作用，假如一架飛機要十個命令來畫出，UNDO祇把最後一個取消，而不可以作一系列UNDO 來除去飛機並恢復背景。

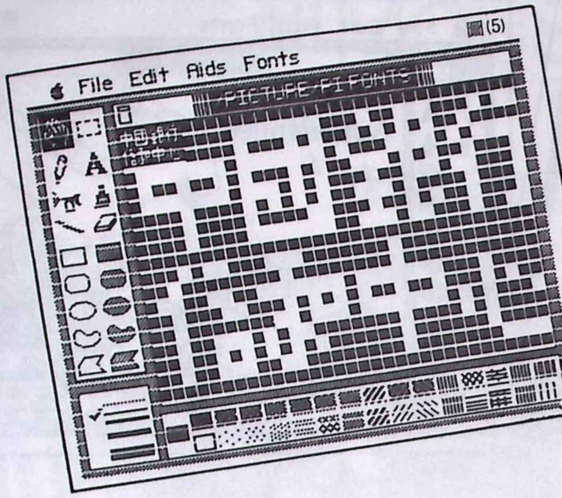
MOUSE PAINT和MAC PAINT一樣，有重疊的視窗（WINDOWS）和以一種RUBBER BANDER 的方式在螢幕上移動。通過RUBBER BANDER 這方式，圖形就會畫出，擦去，然後再在一個鄰近的新位置畫出，造成圖形跟「老鼠仔」方向相同地慢慢拖行，而可以將圓形拉成橢圓，碟形等。至於視窗，它可以同時顯示訊息，調校畫筆的大小（SET BRUSH）和整張繪畫紙的縮圖（SHOW PAGE）。這些視窗是承受了MAC和LISA 的傳統，可以隨意地放在螢幕上的任何一個地方（見圖6）。

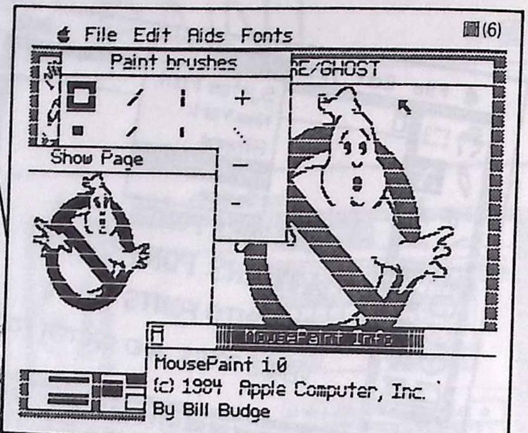
MOUSE PAINT 是一個非常易用和方便的繪圖軟件，不過它仍然有缺點的存在。當用鉛筆來繪畫時，祇有黑白而無彩色。噴霧和畫筆只有黑色。不能塡滿一個圖形，也就是沒有FILL 命令。每次祇可以分割一個圖形。其實MOUSE PAINT 已算是一個很不錯的軟件。

只要你充份地利用MOUSE PAINT 的好處和功能，便可以建立心目中的影像，應用於生產、教育和娛樂各方面，無論對於初學者或是電腦「老手」來說，都是樂趣無窮。好像磁碟月刊第二期的GHOSTBUSTERS ，筆者就是利用「老鼠仔」來繪製它出來的。

MOUSE PAINT 的設計師BILL BUDGE （他是RAS－TER BLASTER和PINBALL CONSTRUCTION SET的作者），爲了滿足用戶需求，盡力將所有功能擠進64K的記憶體，140K 單面磁碟機貯存空間，與及8－BIT的6502微處理器中，因此，你不能奢望48K 的電腦能夠運行它，或是程式會替你CATALOG 你的磁碟。

穿梭蘋果的 APPLE MOUSE 和 MOUSE PAINT





注意，MOUSE PAINT只能支持APPLE'S DOT MATRIX和IMAGE WRITER 這兩部印字機，而且後者只能使用SUPER SERIAL 介面卡（ⅡC 則不需要）。

相信閣下除了喜歡MOUSE PAINT 外，還會喜歡爲「老鼠仔」編寫軟件。幸好那份說明書中，有詳細地介紹如何替你的「老鼠仔」編寫BASIC 和 ASSEMBIY 語言程式。在BASIC 中，只需先打入PRINT CHR＄(4) "PR＃4"（當那張介面卡在SLOT 4），然後打入PRINT CHR＄(1)（設置MOUSE的位置爲零），最後打入PRINT CHR＄(4) "PR＃O"。這時你便可以INPUT資料，如INPUTX，Y，S。X和Y 是「老鼠仔」的位置（0～1024）；S 是按掣的資料，當按下鍵盤，S 值就會是負數了。若果有機會，筆者一定會爲大家編寫一些實用的「老鼠仔」程式，給大家分享。

除了程式設計的指導外，說明書還有教大家如何清潔「老鼠仔」和如何插置介面卡。雖然蘋果公司說介面卡一定要插在SLOT4，但插在其他SLOT 也是沒問題的（除了SLOT 3 外）。

至於將來的MOUSE PAINT 版本，更加會有雙高解像度上繪圖的功能。現時美國已有很多「老鼠仔」軟件，如BEAGLE BROS 的BEAGLE GRAPHIC、BAUDVILLE的BLAZING PADDLES和 BRODERBUND 的DAZZLE DRAW 等。但香港的商場還未出現以上的軟件，相信不久將來，「老鼠仔」是不愁沒有軟件支持的。至於現在可以在高登買到的「老鼠仔」軟件，就只有那套綜合軟件——JANE 及BLAZING PADDLES。■

徵聘特約撰稿員

任何人士有興趣寫作有關CP/M, Z-80, IBM, dBASE II, WORDSTAR, VISICALC或高等語言文章，或有編寫有關程式經驗，歡迎你加入成爲我們的特約撰稿員，待遇從優。

請附同個人學歷／簡歷，連同個人在電腦興趣方面的專長寄來出版社，地址是：九龍中央郵政信箱71193號。

全港最大之蘋果電腦用家組織
招收基本會員

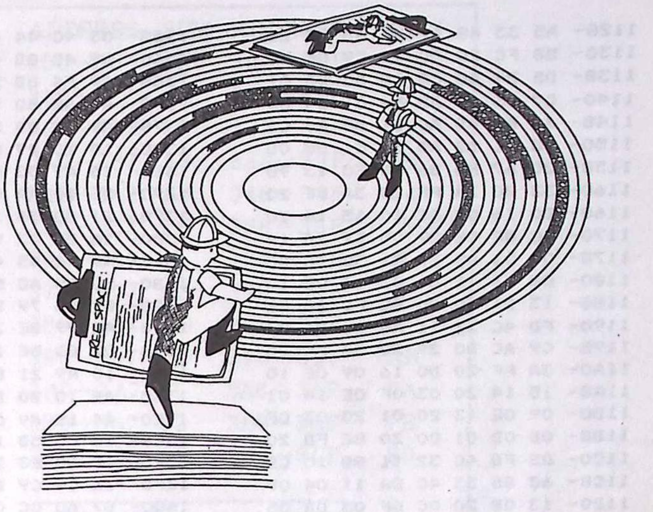
全港最大的APPLE USER 會加入 APPLE FAN CLUB 可令你有意想不到的收獲。

本會軟件庫收集有無數市面有售的軟件供會員借用，此外亦收集有不同種類的軟件使用手册供借閱，歡迎來信索取入會章程。

來信請寄九龍中央郵政信箱 71193 號，信封面請註明〔索取 AFC 章程及入會表格〕，並附同回郵信封一個（貼上四角郵票即可）。

愼防偶然寫花磁碟

磁碟 LOCKER



好多時，我們為了保護一些貴重的磁碟資料，都會貼上WRITE PROTECT紙，防止偶然寫入新資料將原有資料洗掉。可惜有些WRITE PROTECT紙並不好用，而且手頭上又並非經常有這些貼紙。

為了這個原故，筆者設計了一個程式，好等在不需要寫入新資料時，將整張磁碟LOCK起，而到有必要的時候，又可將這張磁碟UNLOCK，讓你重新可再寫入資料。

程式會令磁碟造出一個DISK FULL的假像，好讓瞞騙DOS來過骨。這個磁碟鎖匠（DISK LOCKER）利用一個DUMMY的VTOC來欺騙DOS，當要SAVE 新FILE 入磁碟時，螢幕會打出DISK FULL 字眼。

要造到此效果十分容易，大家只要BRUN DISK LOCK—ER；或BLOAD DISK LOCKER，然後CALL 4100。程式執行時，會有兩項選擇LOCK（鎖住磁碟）或UNLOCK(釋放磁碟)。兩者使用皆非常簡單。不過大家要緊記：任何一隻已經LOCK了或根本就是DISK FULL 的磁碟是不能再下鎖的，將一隻本來沒有LOCK過的磁碟UNLOCK 。如果是強來的話，會嚴重破壞VTOC，此會引出非常嚴重的後果。

當大家要LOCK起一張磁碟時，有幾個步驟要照著去做。首先，要選擇SLOT及DRIVE NO. 然後電腦會將CATA—LOG 讀入緩衝區上。隨著電腦會叫你輸入一個信息，這個信息會以一個反白的TEXT FILE 名字存在於CATALOG之中，開首並且會有DISK LOCKED 這個字眼。一般來說，筆者會用上一個日子，這樣便可以知道這張磁碟有沒有被鎖。

到此，螢幕會讓你選擇是否要將這DISK LOCKER 程式存入你這張做過手脚的磁碟之上。筆者特別有此安排是為了讓大家日後要存放新資料入這張碟時，無須四處尋找這個程式，而且這個程式並不會佔用額外的空間。筆者將它存放入DOS所佔用的三條磁碟中的TRACK 2 上。當一切完成後，電腦會將DUMMY的VTOC 寫入磁碟，而眞正的VT—OC 則被收入TRACK 2 SECTOR 5。最後一個PATCH過的DOS 亦會存入磁碟上。這個PATCH 過的DOS 會防止任何磁碟上的FILE 被DELETE。任何DELETE 的企圖都會得出一個DISK LOCK ERROR的訊號。不過一旦UNL—OCK 該磁碟後，DELETE 就會恢復正常的功能。

要將LOCK了的磁碟解鎖方法非常簡單，大家只要再執行多一次DISK LOCKER ，然後選擇U) NLOCK 功能則可。

本程式是用MERLIN MACRO ASSEMBLER寫成，這裏只列出機械碼，讀者在輸入本程式時，可用下列指令存入磁碟：

BSAVE DISK LOCKER A$1004，L$7AC。

■

```
*******************
**  DISK  LOCKER **
**  A$1004,L$7AD **
*******************
1004- 20 84 FE 20
1008- 2F FB 20 93 FE 20 89 FE
1010- A9 00 85 0A 85 08 85 09
1018- 85 06 A9 02 85 07 20 58
1020- FC 20 D0 16 0C CF C3 CB
1028- A0 CF D2 A0 15 CE CC CF
1030- C3 CB A0 D4 C8 C5 A0 C4
1038- C9 D3 CB BF A0 00 20 6F
1040- FD AD 00 02 C9 CC D0 03
1048- 4C 63 10 C9 D5 D0 CF 4C
1050- 86 14 01 00 00 00 00 00
1058- FB B7 00 00 00 00 00 00
1060- 00 00 00 A9 00 85 02 85
1068- 04 85 05 A9 2F 85 03 20
1070- 00 16 20 FF 16 A2 00 A9
1078- 0B 85 01 A4 01 B1 02 C9
1080- 02 D0 02 E6 0A C9 FF F0
1088- 08 C9 00 D0 17 A9 FF 91
1090- 02 98 85 04 A5 03 85 05
1098- A5 09 D0 08 A5 04 85 08
10A0- A5 05 85 09 18 A5 01 69
10A8- 23 85 01 E8 E0 07 F0 03
10B0- 4C 7B 10 C6 03 A5 03 C9
10B8- 20 F0 03 4C 75 10 A0 FF
10C0- B1 02 D0 08 88 C0 37 D0
10C8- F7 4C D0 10 A5 05 D0 37
10D0- 20 3A FF 20 3A FF 20 8E
10D8- FD 20 D0 16 44 49 53 4B
10E0- 60 4C 4F 43 4B 45 44 60
10E8- 4F 52 60 46 55 4C 4C 6E
10F0- 60 4F 50 45 52 41 54 49
10F8- 4F 4E 60 41 42 4F 52 54
1100- 45 44 6E 00 4C D6 14 A9
1108- 11 A0 00 91 04 C8 C8 A9
1110- 00 91 04 A5 09 C5 05 D0
1118- 08 A5 08 C5 04 D0 02 E6
1120- 0A 18 A5 04 69 03 85 04
```

```
1128- A5 33 48 A9 BF 85 33 20
1130- 58 FC 20 D0 16 C9 CE D0
1138- D5 D4 A0 CD C5 D3 D3 C1
1140- C7 C5 A0 A8 C4 C1 D4 C5
1148- A9 A0 AD A0 B1 B8 A0 C3
1150- C8 C1 D2 A0 CD C1 D8 00
1158- 20 67 FD 86 00 E0 13 90
1160- 33 20 3A FF 20 3A FF 20
1168- D0 16 09 0E 10 15 14 20
1170- 0C 0F 0E 07 05 12 20 14
1178- 08 01 0E 20 31 38 20 03
1180- 08 01 12 01 03 14 05 12
1188- 13 21 00 20 8E FD 20 8E
1190- FD 4C 32 11 A4 00 B1 06
1198- C9 AC D0 29 20 3A FF 20
11A0- 3A FF 20 D0 16 09 0E 10
11A8- 15 14 20 03 0F 0E 14 01
11B0- 09 0E 13 20 01 20 03 0F
11B8- 0D 0D 01 00 20 8E FD 20
11C0- 8E FD 4C 32 11 88 10 CE
11C8- 68 85 33 4C DA 11 04 09
11D0- 13 0B 20 0C 0F 03 0B 05
11D8- 04 20 A0 1D A9 A0 91 04
11E0- 88 10 FB A0 1E A9 00 91
11E8- 04 C8 91 04 A0 0B B9 CE
11F0- 11 91 04 88 10 F8 18 A5
11F8- 04 69 0C 85 04 A5 05 69
1200- 00 85 05 A4 00 88 B1 06
1208- 29 3F 91 04 88 10 F7 A5
1210- 0A F0 03 4C E8 12 20 58
1218- FC 20 D0 16 D3 C1 D6 C5
1220- A0 D4 C8 C9 D3 A0 D0 D2
1228- CF C7 D2 C1 CD BF A0 A8
1230- D9 AF CE A9 A0 00 20 6F
1238- FD AD 00 02 C9 D9 F0 3A
1240- C9 CE D0 D2 4C E8 12 02
1248- 0F 84 C1 D0 D0 CC C5 A0
1250- C4 C9 D3 CB A0 CC CF C3
1258- CB A0 A0 A0 A0 A0 A0 A0
1260- A0 A0 A0 A0 A0 A0 A0 A0
1268- 09 00 02 07 02 08 02 09
1270- 02 0A 02 0B 02 0C 02 0D
1278- 02 0E A0 00 B9 47 12 91
1280- 08 C8 C0 23 D0 F6 A0 FF
1288- A9 18 85 03 A9 00 85 02
1290- 91 02 88 C0 FF D0 F9 A9
1298- 0C 85 02 B9 6A 12 91 02
12A0- C8 C0 10 D0 F6 A9 04 8D
12A8- 00 10 A9 10 8D 01 10 A9
12B0- F0 8D 02 10 A9 07 8D 03
12B8- 10 A9 18 8D 5B 10 A9 02
12C0- 8D 5E 10 8D 56 10 A9 0F
12C8- 8D 57 10 A0 52 A9 10 20
12D0- D9 03 90 03 4C 44 15 A9
12D8- 00 85 48 CE 57 10 CE 5B
12E0- 10 AD 57 10 C9 06 D0 E3
12E8- 20 8A 17 A9 00 85 02 A9
12F0- 20 85 03 A0 FF A9 00 91
12F8- 02 88 C0 37 D0 F7 20 39
1300- 17 A9 01 8D 5E 10 8D 56
1308- 10 8D 57 10 A9 20 8D 5B
1310- 10 A0 52 A9 10 20 D9 03
1318- 90 03 4C 44 15 A9 00 85
1320- 48 A9 AF 8D 66 A2 8D 66
1328- 20 A9 AE 8D 67 A2 8D 67
1330- 20 A9 02 8D 5E 10 A0 52
1338- A9 10 20 D9 03 90 03 4C
1340- 44 15 A9 00 85 48 A9 01
1348- 8D 5E 10 A9 04 8D 57 10
1350- A0 52 A9 10 20 D9 03 90
1358- 03 4C 44 15 A9 00 85 48
1360- A9 4C 8D 4F 20 8D 4F A5
1368- A9 F4 8D 50 20 8D 50 A5
1370- A9 AE 8D 51 20 8D 51 A5
1378- A9 02 8D 5E 10 A0 52 A9
1380- 10 20 D9 03 90 03 4C 44
1388- 15 A9 00 85 48 A9 01 8D
1390- 5E 10 A9 0E 8D 57 10 A9
1398- 21 8D 5B 10 A0 52 A9 10
13A0- 20 D9 03 90 03 4C 44 15
13A8- A9 00 85 48 CE 57 10 CE
13B0- 5B 10 AD 5B 10 C9 1F D0
13B8- E3 A0 79 B9 0C 14 99 8E
13C0- AE 99 8E 20 88 10 F4 A9
13C8- 02 8D 5E 10 A9 0E 8D 57
13D0- 10 A9 21 8D 5B 10 A0 52
13D8- A9 10 20 D9 03 90 03 4C
13E0- 44 15 A9 00 85 48 CE 57
13E8- 10 CE 5B 10 AD 5B 10 C9
13F0- 1F D0 E3 20 58 FC 20 D0
13F8- 16 C4 C9 D3 CB A0 CE CF
1400- D7 A0 CC CF C3 CB C5 C4
1408- 00 4C D6 14 60 01 FF FF
1410- 00 02 05 FB B7 BB B3 00
1418- 00 01 00 00 FF FF D2 CF
1420- D2 D2 C5 A0 CB C3 CF CC
1428- A0 CB D3 C9 C4 AD 6A AA
1430- 0A 0A 0A 0A 8D 90 AE 8D
1438- 9E AE AD 68 AA 8D 91 AE
1440- 8D 9F AE 20 E3 03 84 00
1448- 85 01 A0 10 B9 8F AE 91
1450- 00 88 10 F8 A4 00 A5 01
1458- 20 D9 03 90 08 20 8E FD
1460- A2 08 4C 02 A7 AD BC B3
1468- D0 08 A9 05 20 AA A2 4C
1470- 68 A2 20 8E FD 20 3A FF
1478- A0 0E B9 A0 AE 20 ED FD
1480- 88 10 F7 4C D0 03 20 00
1488- 16 A9 01 8D 5E 10 A9 02
1490- 8D 56 10 A9 05 8D 57 10
1498- A9 20 8D 5B 10 20 73 17
14A0- A0 52 A9 10 20 D9 03 90
14A8- 03 4C 44 15 A9 00 85 48
14B0- AD 01 20 C9 11 F0 22 20
14B8- 3A FF 20 3A FF 20 8E FD
14C0- 20 D0 16 44 49 53 4B 60
14C8- 49 53 60 4E 4F 54 60 4C
14D0- 4F 43 4B 45 44 00 4C D0
14D8- 03 A9 21 8D 2C 17 20 FF
14E0- 16 A9 20 8D 2C 17 A9 00
14E8- 85 02 AA A9 2F 85 03 A9
14F0- 0B 85 01 A4 01 B1 02 C9
14F8- 11 D0 07 A9 FF 91 02 4C
1500- 1E 15 18 A5 01 69 23 85
1508- 01 E8 E0 07 F0 03 4C F3
1510- 14 C6 03 A5 03 C9 20 F0
1518- 05 A2 00 4C EF 14 20 39
1520- 17 A9 00 8D 01 20 20 8A
1528- 17 20 58 FC 20 D0 16 C4
1530- C9 D3 CB A0 CE CF D7 A0
1538- D5 CE CC CF C3 CB C5 C4
1540- 00 4C D6 14 20 8E FD A9
1548- 00 85 48 AD 5F 10 C9 10
1550- D0 32 20 3A FF 20 D0 16
1558- 04 09 13 0B 20 17 12 09
1560- 14 05 20 10 12 0F 14 05
1568- 03 14 05 04 20 2D 20 0F
1570- 10 05 12 01 14 09 0F 0E
1578- 20 01 02 0F 12 14 05 04
1580- 00 4C D6 14 C9 40 D0 13
1588- 20 3A FF 20 3A FF 20 D0
1590- 16 04 12 09 16 05 20 00
1598- 4C DF 15 C9 80 D0 12 20
15A0- 3A FF 20 3A FF 20 D0 16
15A8- 12 05 01 04 20 00 4C DF
15B0- 15 20 3A FF 20 3A FF 20
15B8- D0 16 CF D0 C5 D2 C1 D4
15C0- C9 CF CE A0 C1 C2 CF D2
15C8- D4 C5 C4 A0 AD A0 C5 D2
15D0- D2 CF D2 A0 A4 00 AD 5F
15D8- 10 20 DA FD 4C D6 14 20
15E0- D0 16 05 12 12 0F 12 20
15E8- 2D 20 0F 10 05 12 01 14
15F0- 09 0F 0E 20 01 02 0F 12
15F8- 14 05 04 2E 00 4C D6 14
1600- A9 00 85 25 85 24 20 22
1608- FC 20 42 FC 20 D0 16 C5
1610- CE D4 C5 D2 A0 D3 CC CF
1618- D4 A0 A3 A0 C4 C5 C6 C1
1620- D5 CC D4 A0 BD A0 00 AD
1628- 6A AA 20 DA FD A9 18 85
1630- 24 20 6F FD A0 00 B1 06
1638- C9 8D D0 17 A9 00 85 25
1640- A9 17 85 24 20 22 FC AD
1648- 6A AA 20 DA FD 20 8E FD
1650- 4C 66 16 C9 B8 90 03 4C
1658- 00 16 C9 B1 B0 03 4C 00
1660- 16 29 0F 8D 6A AA A9 02
1668- 85 25 A9 00 85 24 20 22
1670- FC 20 42 FC 20 D0 16 C5
1678- CE D4 C5 D2 A0 C4 D2 C9
1680- D6 C5 A0 A3 A0 C4 C5 C6
1688- C1 D5 CC D4 A0 BD A0 00
1690- AD 68 AA 20 DA FD A9 19
1698- 85 24 20 6F FD A0 00 B1
16A0- 06 C9 8D D0 17 A9 02 85
16A8- 25 A9 18 85 24 20 22 FC
16B0- AD 68 AA 20 DA FD 20 8E
16B8- FD 4C CF 16 C9 B3 90 03
16C0- 4C 66 16 C9 B1 B0 03 4C
16C8- 66 16 29 0F 8D 68 AA 60
16D0- 68 8D E1 16 68 8D E2 16
16D8- EE E1 16 D0 03 EE E2 16
16E0- AD FF FF F0 06 20 ED FD
16E8- 4C D8 16 18 AD E1 16 69
16F0- 01 8D FD 16 AD E2 16 69
16F8- 00 8D FE 16 4C FF FF A9
1700- 2F 8D 5B 10 A9 0F 8D 57
1708- 10 A9 11 8D 56 10 A9 01
1710- 8D 5E 10 20 73 17 A0 52
1718- A9 10 20 D9 03 90 05 68
1720- 68 4C 44 15 A9 00 85 48
1728- AD 5B 10 C9 20 F0 09 CE
1730- 57 10 CE 5B 10 4C 16 17
1738- 60 A9 11 8D 56 10 A9 0F
1740- 8D 57 10 A9 2F 8D 5B 10
1748- A9 02 8D 5E 10 20 73 17
1750- A0 52 A9 10 20 D9 03 90
1758- 05 68 68 4C 44 15 A9 00
1760- 85 48 CE 57 10 CE 5B 10
1768- AD 5B 10 C9 1F F0 03 4C
1770- 50 17 60 AD 68 AA 8D 54
1778- 10 8D 62 10 AD 6A AA 0A
1780- 0A 0A 0A 8D 53 10 8D 61
1788- 10 60 A9 02 8D 56 10 A9
1790- 05 8D 57 10 A9 02 8D 5E
1798- 10 A9 20 8D 5B 10 A0 52
17A0- A9 10 20 D9 03 90 05 68
17A8- 68 4C 44 15 A9 00 85 48
17B0- 60
```

VTOC 編輯器

黃偉江

ADDR	USES
$06	GENERAL
$300	TRACK NO.
$301	SECTOR NO.
$302	GENERAL
$303	BASE
$304	RWTS
$30C	IOB
$322	PRINT TRACK
$348	ASCII "*"
$34D	ASCII "·"
$36E	PRINT MAP
$381	ZAP BYTE
$1800	VTOC BUFFER

圖一：副程式及其他

ADDRESS	SUM	ADDRESS	SUM
300—307	91	308—30F	BE
310—317	4A	318—31F	61
320—327	C9	328—32F	80
330—337	E3	338—33F	0B
340—347	A0	348—34F	B4
350—357	0E	358—35F	2E
360—367	12	368—36F	97
370—377	C3	378—37F	67
380—387	A9	388—38F	89
390—397	3D	398—39F	AB
3A0—3A7	21	3A8—3AF	5A
3B0—3B7	02	3B8—3BF	7E
3C0—3C7	5D	3C8—3CF	E0

圖二：CHECK SUM核對表

VTOC（TRACK 17，SECTOR 0）掌管了整片磁碟的重要資料，包括了DOS 版本、目錄表所在位置、磁碟號碼，以及最重要的BIT MAP。

各位或會發覺到有時需要更改磁碟上VTOC 的內容，例如當要釋放出更多的空間，或是要LOCK 起某些已壞了的SECTOR 等等。凡此種種皆要由一些DISK ZAP 之類的程式READ進TRACK 17，SECTOR 0 後，繼而鍵入所需SECTOR 的狀態。其間需要到二進制及十六進制數字的轉換，總覺十分麻煩。

及後筆者看到了一些DISK MAP 程式，想到了如果能夠在一幅DISK MAP 之中用游標上下左右隨意把某個SEC—TOR LOCK或UNLOCK，豈不方便快捷？以下介紹的一個程式便是特此而設計。

程式特點

40TRACK 磁碟日漸普遍，故本程式特別適用於35／40 TRACK 的磁碟，而且程式更能自動分辨。因爲APPLE 的TEXT SCREEN 只得40COLUMN ，顯示了40TRACK 的內容後便沒有位再顯示SECTOR NUMBER。故此本程式的游標乃是代表該SECTOR NUMBER 的反相顯示，所以游標的數字便是該SECTOR的NUMBER，這樣便一目了然了。以下便是本程式所用的指令：

使用指令

I——I、J、K、M 爲游標控制鍵。I是把游標移上一SEC—TOR 。

J——把游標移左一TRACK（即減少TRACK NUMBER）

K——把游標移右一TRACK。

M——把游標下移一SECTOR。

L——把游標目前所處的TRACK和SECTOR LOCK起。

U——把游標目前的TRACK和SECTOR無論有沒有使用都變爲空SECTOR。

R——把磁碟上的VTOC讀進了主機，跟着在螢幕上印出了一幅DISK MAP ，最後就在左下角顯示該碟是35還是40 TRACK ，以及在右下角印出該碟的VOLUMN NUMBER。

CTRL—W——把更改後的VTOC寫回磁碟裏。筆者選用CTRL—W 而不用W是因爲避免了一時手多或不小心亂按以招致嚴重的後果。

CTRL—Q——清理畫面，離開本程式。

程式原理

本程式分爲了兩部份，BASIC 主程式以及機械語言副程式MAP. OBJ（A$300，L$D0）。BASIC 部份只要負責了輸出輸入之類，重要的如DISK MAP 印出或更改內容都交予機械語處理，以取其快捷。

筆者用了*代表了LOCK起的SECTOR ，而·代表了空SECTOR ，如讀者不喜歡可參看圖一更改。MAP.OBJ內的副程式用途都一同列在圖一裏，以供各位參考。

MAP.OBJ 的鍵入可以用筆者所作的MCIE （機械碼輸入編輯器），詳細使用方法可參考十五期拙作。圖二爲MAP.OBJ的CHECK SUM 表，各位可以一面打入一面核對，以避免錯誤。

VTOC 內資料格式相信讀者已知道不少，而本刊以前亦介紹過不少，故筆者不擬詳述。最後一點要提及的是更改後的DISK MAP，是要按CTRL—W 寫回磁碟中，才能作實。

本文共有兩個程式(1)VTOC EDITOR及MAP.OBJ 後者於打入後，可用下列指令存入磁碟：BSAVE，A$300，L$D0 ■

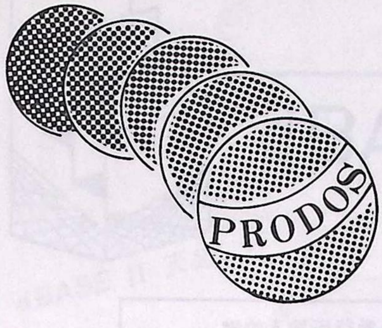
VTOC編輯器

```
10  REM  *******************
20  REM  *                 *
30  REM  *   VTOC  EDITOR  *
40  REM  *                 *
50  REM  * BY WONG WAI KONG *
55  REM  * HKTKPC    23-2-85 *
60  REM  *                 *
70  REM  *******************
99  REM
100  REM  -INITIALIZE-
110  LOMEM: 8192
120  IF  PEEK (772) <  > 169 THEN
     PRINT  CHR$ (13): PRINT  CHR$
    (4)"BLOAD MAP.OBJ,A$300"
130 T = 0:S = 0:H$ = "0123456789A
    BCDEF":N = 34
140 RWTS = 772:TRACK = 802:MAP =
    878:ZAP = 897:NA = 6196:VA =
    6150:CLR = 64668:Z = 944:K =
     - 16384
150  TEXT : HOME : FOR I = 1 TO 1
    6: PRINT "0";: NEXT : FOR I =
    1 TO 16: PRINT "1";: NEXT : FOR
    I = 1 TO 8: PRINT "2";: NEXT

170  VTAB 2: PRINT H$;H$;: PRINT
    "01234567"
180  VTAB 3: INVERSE : FOR I = 1 TO
    40: PRINT " ";: NEXT
190  VTAB 20: FOR I = 1 TO 40: PRINT
    " ";: NEXT
200  VTAB 3: PRINT "TRACK :";: HTAB
    30: PRINT "VTOC EDITOR"
210  VTAB 20: HTAB 4: PRINT "TRAC
    KS";: HTAB 26: PRINT "DISK V
    OLUME=": NORMAL
1000  REM  -MAIN LOOP-
1010  FOR LOOP = 1 TO 2 STEP 0
1020  HTAB T + 1: VTAB S + 4: INVERSE
    : PRINT  MID$ (H$,S + 1,1): NORMAL
1025  WAIT K,128:KBD =  PEEK (K) -
    128: POKE  - 16368,0
1030  POKE 768,T: CALL TRACK
1032  IF KBD = 73 THEN S = S - 1
1034  IF KBD = 74 THEN T = T - 1
1036  IF KBD = 75 THEN T = T + 1
1038  IF KBD = 77 THEN S = S + 1
1040  IF KBD = 82 THEN  GOSUB 110
    0
1042  IF KBD = 23 THEN  GOSUB 120
    0
1050  IF KBD = 76 THEN  POKE Z,19
    1: GOSUB 2000
1055  IF KBD = 85 THEN  POKE Z,17
    8: GOSUB 2000
1080  IF KBD = 17 THEN  TEXT : HOME
    : END
1090  IF S > 15 THEN S = 0:T = T +
    1
1093  IF S < 0 THEN S = 0:T = T -
    1
1095  IF T > N THEN T = N
1097  IF T < 0 THEN T = 0
1099  NEXT
1100  REM  -READ VTOC-
1110  POKE 793,1: CALL RWTS: POKE
    72,0: CALL MAP
1140 N =  PEEK (NA) - 1:V =  PEEK
    (VA): VTAB 20: HTAB 1: INVERSE
    : PRINT N + 1;: HTAB 38: PRINT
    V: NORMAL
1150  RETURN
1200  REM  -WRITE VTOC-
1210  POKE 793,2: CALL RWTS: POKE
    72,0: RETURN
2000  REM  -ZAP-
2010  POKE 768,T: POKE 769,S: CALL
    ZAP: CALL TRACK:S = S + 1: RETURN
3000  REM THANKS TO LO KWOK KEE
3010  REM           LEE CHUN WAN
3020  REM           FAN KAN SHING
```

```
*****************
**  MAP. OBJ  **
** A$300,L4D0 **
*****************

0300- 00 00 00 38 A9 03 A0 0D
0308- 20 D9 03 60 00 01 60 01
0310- 00 11 00 1E 03 00 18 00
0318- 00 01 00 FE 60 01 00 01
0320- EF D8 AD 00 03 85 24 A9
0328- 12 85 25 AD 00 03 0A 0A
0330- 18 6D 03 03 8D 3B 03 8D
0338- 60 03 AD 38 18 85 06 20
0340- 22 FC 18 06 06 B0 05 A9
0348- AA 4C 4E 03 A9 AE 20 F6
0350- FD 20 20 FC 20 10 FC A9
0358- 0A C5 25 D0 0A A2 01 BD
0360- 38 18 85 06 4C 3F 03 A9
0368- 02 C5 25 D0 D2 60 A9 00
0370- 8D 00 03 20 22 03 EE 00
0378- 03 A9 28 CD 00 03 D0 F3
0380- 60 AD 00 03 0A 0A 18 6D
0388- 03 03 8D AD 03 8D B6 03
0390- 8D CC 03 AD 01 03 29 07
0398- AA A9 01 E0 00 D0 1A 8D
03A0- 02 03 AD 01 03 A2 00 C9
03A8- 08 10 01 E8 BD 38 18 4C
03B0- 00 03 0D 02 03 9D 38 18
03B8- 60 18 0A CA 4C 9B 03 48
03C0- A9 FF 38 ED 02 03 85 06
03C8- 68 25 06 9D 38 18 60 00
```



UTILITY 磁碟使用方法

在上期的 PRODOS 系統一文中我曾經向各位介紹過 UTILITY DISK ，它是一套給予 APPLE 用戶的發展系統，APPLE 公司稱之爲 DEVELOPER SYSTE—M，它的目錄已在上期刊登過，主要的功能是提供機械語言及 APPLESOFT 在 APPLE 上的應用，雖然到目前爲止我們還沒有 MANUAL，不過現在我將一些使用心得和各位一同分享。

首先我將它的目錄再次列出（圖1），這個「發展系統」只有一面磁碟，VOLUME 名字是／PRODOS／，其上的檔案可以分爲兩種，第一種是 APPLE 公司允許用戶將這些檔案隨同自己編寫的程式出售，第二種的檔案是公用程式，只可作私人應用，不可以轉讓給別人。

第一種類型的檔案「授權使用」的，它們共有4個：

① PRODOS ——這個檔案的類型是以系統的形態出現（SYS），由一系列的機械語言副程式組成，在上期介紹過的 PRODOS TECHNICAL REFERENCE M—ANUAL 內各位可以找到具體的結構資料，從而配合發展機械語言程式。

② BASIC・SYSTEM——這個系統程式的主要功能是配合 PRODOS系統檔案，使 APPLESOFT 程式能夠在 PRODOS 系統中指揮磁碟機作資料存取。在上期介紹過的 BASIC PROGRAMMING WITH PRO—DOS 這本書內有詳細的使用方法說明，而較深入的技術資料也可以在 TECHNICAL MANUAL 中找到。

③ CONVERT ——這個系統程式是 DOS 與 PRODOS 之間作檔案轉換之時使用，在上期已經詳細討論過它的使用方法及應用。

④ FILER——這個系統程式的功能是包括了：(A)格式化 PRODOS 磁碟(B)抄錄檔案(C)檔案組織，各位可以參看 P—RODOS USER MANUAL 第 19 至 95 頁有關它的詳細使用方法。

第二種類型的公用程式檔案包括有：

①EXERCISER——這是一個系統程式，螢幕如圖2所示，有關它的使用方法可以在 PRODOS TECHNICAL REFERENCE MANUAL 中找到，這個 EXERCIS—ER 是需要 PRODOS的 BASIC 存在於記憶中才可以運作，你只要鍵入一 EXERCISER 便可以執行它，而這個 EXERCISER 開始時會佔用記憶體中由$2000開始的一段位置，之後會移往 $7A00 開始的記憶體位址。但我發現它並沒有設立 HIMEM 來作保護，因此若果各位參考 TECHNICAL MANUAL 使用它時，要緊記不要將緩衝區域放置到程式之上的位置。

②BUGBYTER ——這個程式是用來將機械語言程式作除錯之用，它可以：(A)執行一個單獨的機械碼指令，(B)程式循跡，(C)設立程式的中斷點以測試或檢驗其結果，(D)滙編程式及(E)反組譯機械碼。從 TECHNICAL MANUAL 中我發覺它在運作時會直接呼叫 PRODOS的核心（KE—RNEL）網路，故此 BUGBYTER 可以直接使用磁碟機來存取資料，不過它的使用方法是需要在 PRODOS ASSEMBLER TOOLS MANUAL 中才可以找得到，將來若果找到有關資料會在此再介紹給各位。

③EDASM・SYSTEM和EDASM・ED 及 EDASM・ASM ——這三個檔案包括了在 PRODOS 系統上的滙編語言編輯器及組譯器，事實上這套系統是 DOS 3.3 的 EEASM 的改良 PRODOS版本。使用方法和昔日的 EDASM 相似，不過執行之前須先設定 PREFIX，你亦可以鍵入 ONLINE 命令來顯示 VOLUME，而鍵入？可以將 EDASM 的全部指令列出到螢幕上。

④APA ——這個檔案是以往的 APPLESOFT PROGR—AMMERS ASSISTANT 的 PRODOS 改良版本，可以用作 APPLESOFT 程式設計的輔助工具，你可以從 BASIC PROGAMMING WITH PRODOS 中的 APPENDIX D 找到它的使用方法。

⑤ABLE ——這個程式的全名是 APPLESOFE BASI—C LINE EDITOR 的縮寫，執行這個程式的方法很簡單，只需鍵入 BRUN ABLE 。使用它來編輯行句時可以鍵入〝EDIT <行號>〞，例如EDIT 10即可編輯 BASIC 程式的第10行。若要中止 ABLE 可以執行 APA 程式，以下是編輯模式時的命令：

CTRL—D 刪除行句中游標所在位置的字元。
CTRL—I 在游標所在位置加插字元到行句之中。
CTRL—R 將編輯前的行句內容重現。
CTRL—B 將游標移到行句之開始位置上。
CTRL—E 將游標移到行句之末端位置。
CTRL—T 刪除行句中游標位右邊的內容。

在編輯模式中，無論游標在甚麼位置之上，只要鍵入 RETURN 鍵即可接受整個行句的輸入，無須再將游標以右箭咀移到末端，因此這是一個很好的公用程式。

以上爲各位介紹了各個系統程式的功用及一些使用心得，現在爲各位作一個使用這面磁碟的方法的總結，希望各位可從 TECHICAL MANUAL和其他已介紹的資料中獲得更多有關這面UTILITY 磁碟的用途及使用心得。

當這面磁碟被啓動之時，上面的 STARTUP 程式會先顯示出你的電腦系統的週邊咭所在位置，之後會自動進入

UTILITY 磁碟使用方法

BASIC的命令模式，這個時候你可以鍵入"—"鍵來執行上面所述的系統程式：

程式名稱	執行時按入的鍵	中止執行所按入的鍵
FILER	—FILER	Q
CONVERTER	—CONVERT	Q
EDASM	—EDASM·SYSTEM	EXIT (PATHNAME)
EXERCISER	—EXERCISER	Q
BUGBYTER	BRUN BUGBYTER ;A$ (XXXX)	Q
BASIC	—BASIC·SYSTEM	— (PATHNAME)
APA	—APA	EXIT
ABLE	BRUN ABLE	

```
NAME              TYPE  BLOCKS  MODIFIED              CREATED                 ENDFILE SUBTYPE

*PRODOS           SYS      31    1-JAN-84   0:00     <NO DATE>                  15360
*BASIC.SYSTEM     SYS      21   15-NOV-83   0:00     <NO DATE>                  10240
*CONVERT          SYS      42    1-NOV-83   0:00     <NO DATE>                  20481
*FILER            SYS      51   <NO DATE>            <NO DATE>                  25600
 STARTUP          BAS       7   16-NOV-83  15:57     <NO DATE>                   2699
*EDASM.SYSTEM     SYS       9   14-DEC-83  15:32     <NO DATE>                   4095
*EDASM.ED         BIN      17   14-DEC-83  15:32     <NO DATE>                   8191  A=$3000
*EDASM.ASM        BIN      29   14-DEC-83  15:26     <NO DATE>                  14079  A=$3000
*RBOOT            BIN       1   14-DEC-83  15:35     <NO DATE>                    410  A=$0218
*RLOAD            BIN       3   14-DEC-83  15:36     <NO DATE>                    661  A=$0800
*EXERCISER        SYS      16   17-OCT-83   0:00     <NO DATE>                   7385
*BUGBYTER         BIN      16   15-DEC-83   0:00     <NO DATE>                   7177  A=$2000
*FORMATTER        TXT       3   20-DEC-83   0:00     20-DEC-83    0:00            764  R=    0
*APA              BIN      10   11-NOV-83   7:37     <NO DATE>                   4608  A=$2000
*BYE              BAS       1   16-NOV-83  15:55     <NO DATE>                    158
*ABLE             BIN       4   21-NOV-83  11:10     <NO DATE>                   1060  A=$2000

BLOCKS FREE:   12     BLOCKS USED:   268          TOTAL BLOCKS:   280
```

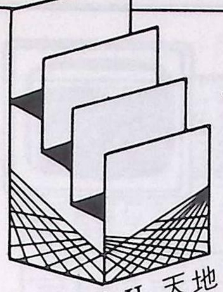
圖(1)

```
* * * * * * * * * * * * * * * * * * * * * *
*                 PRODOS                   *
*     MACHINE LANGUAGE INTERFACE           *
*        SYSTEM CALL EXERCISER             *
* * * * * * * * * * * * * * * * * * * * * *

$C0-CREATE                 $CB-WRITE
$C1-DESTROY                $CC-CLOSE
$C2-RENAME                 $CD-FLUSH
$C3-SET FILE INFO          $CE-SET MARK
$C4-GET FILE INFO          $CF-GET MARK
$C5-ON LINE                $D0-SET EOF
$C6-SET PREFIX             $D1-GET EOF
$C7-GET PREFIX             $D2-SET BUF
$C8-OPEN                   $D3-GET BUF
$C9-NEWLINE                $80-READ BLOCK
$CA-READ                   $81-WRITE BLOCK

L - LIST DIRECTORY         Q - QUIT
M - MODIFY BUFFER
```

圖(2)



dBASE II 天地

dBASE

入門篇

編寫程式的第一步

在上期我們對DBASE的入門以最簡潔最直接的途徑介紹了以交談形式（INTERACTIVE）來應用它，所謂交談形式是指你在點命令模式之中對DBASE下一度指令，它便為你做一件工作。不過我們所介紹的DBASE指令是很有限的，若果要充份掌握DBASE功能（指令），最好參考DBASE的USER MANUAL 。這是DBASE原作者WAYNE RATLIFF 所編著的，全書共分為五個部份，第一及第二個部份是與DBASE作直接交談形式使用的方法及指令。

現在我要假定各位朋友都有熱誠的意願去掌握DBASE，因此在學習中的第一個階段先要掌握DBASE的指令及句法，在USER MANUAL的第一部份裏及上期我所介紹的資料上你可以了解及總結DBASE這個關連式資料庫管理系統所能提供給你的服務能力，並且你可以使用CREATE 命令來建立一個新的資料庫並開始輸入資料。

假使你想要更改資料，你可以用EDIT，DELETE，RECALL及PACK等命令。你也可以用APPEND及INSERT 命令來加入所需的更多資料，甚至你可以用LIST及DISPLAY 命令將整個檔案或選擇幾項特別的記錄將它列出來。USER MANUAL的第一部份亦介紹了使用GOTO 及SKIP 命令在一資料庫之中容易地而且迅速地移動記錄。除此以外，DBASE Ⅱ也可以用？命令來當成強有力的交談式計算器使用，作者也曾介紹過使用運算式的方式及如何展延DBASE 命令被使用的能力。

在USER MANUAL的第二個部份裏作者對使用運算式的方法及如何展延DBASE命令的使用能力作有更深入的探討，使你在閱讀之後可以更容易及更迅速取出有用的資料。

在USER MANUAL 的第二個部份裏作者已告訴你有多少種不同的運算式（ARITHMETIC，RELATIONAL 及STRING形式）可以被用來修改DBASE的命令，使你能夠比其他資料庫管理系統具有更深度的控制資料能力。

因為資料結構是資料庫系統的根本，作者在第二部份裏曾介紹許多種不同的方法來使你可以修改資料結構，也指引了很多思考角度給你作考慮是否要使用或不使用資料庫內的資料。他還說明了如何進入，改變和找尋想要的特別資料的方法，介紹了新的全域性（GLOBAL）命令，使你能夠用單一的命令如（COUNT，SUM，REPORT，TOTAL 等）將所有的資料轉換成為不同環境內需要的資訊，例如用文字處理器來處理等。

到這裏為止，你應該自我檢討一下以上各點你是否已充份掌握。若果你已做到的話，也了解如何去寫一個運算式，你已經非常接近能編寫程式了，這是學習DBASE的第二個階段。所謂程式是將一連串的指令放在一起，到需要時叫電腦照著它逐步去做。

要建立一個命令檔（即程式），你可以列出你所要執行的CP／M 檔案的命令，並且可用編輯程式（TEXT EDITOR）或文字處理器如WORDSTAR等在其名稱後面加上.CMD 展延檔名。

現在讓我們建一個命令檔叫作RECORDS，當然你可以使用編輯程式或文字處理器來做這件事，但是DBASE提供一個更簡單的辦法，你只要在進入DBASE後鍵入MODIFY COMMAND RECORDS，這時DBASE會展示一個空白的螢幕，在上面你可以使用先前（上期）提到過的全螢幕編輯特性將你要的命令寫進去。現在鍵入如下：

```
USE B:NAME

INDEX ON NAME TO B:INDNAME

USE B:NAME INDEX B:INDNAME

SET TALK OFF

GOTO TOP

SET PRINT ON

DO WHILE .NOT. EOF

    ? NAME
    ? TEL
    ?

  SKIP

ENDDO

SET PRINT OFF
```

編寫程式的第一步

```
1. USE THE PHONE LIST FILES DATA ON DRIVE B

2. INDEX THE DATA BY FOCUSING TO THE FIELD CALLED NAME,SUCH THAT
   THE NAME WILL BE APPEARED ALPHABETICALLY.

3. A LOOP TO PRINT OUT THE NAME & TEL IN THE FILE.
```

圖(1)

假如你已經照著上面的程式鍵入時，或者在將來編寫更具規模的程式（命令檔），你會發覺到DBASE的內建編輯功能是一種最方便的利器，因爲你可以直接寫一修改程式而不須回到電腦的系統階層裏去做便可以達到，而你最後僅需鍵入CTRL－W便可將這程式貯存到碟上。

到目前爲止，若要執行這個程式，只要在DBASE之中鍵入DO PHONE 即可。我會在稍後解說這個程式的目的及功能，但我先要告訴各位一些有關程式設計時需要注意的事項。

DBASE的原作者RATLIFF 曾經說過當你想要建立一個命令檔時，第一件要做的事是將電腦關掉。這是正確的；因爲這也是許多朋友常犯的錯誤，他們甚至在尙未有"要做甚麽"的清楚概念之前，就立即開始在電腦上開始鍵入東西。既然我們明知這種現象出現我們就得去更正。

最好的方法是先了解在許多書中所說有關結構化程式設計的方法及結構化語言的使用。簡單地說，下面是最容易掌握的方法：

首先以普通英文來定義你的問題，使它成爲一般說明，就以我們的程式爲例，請參閱圖1。

再深入一點定義，究竟你需要輸入些甚麽？及你想要的輸出和報表的格式是怎樣？

下一步，看看有甚麽例外。甚麽是起始條件？如果記錄消失了會出現甚麽情況？

一旦你已定義好你想要做的，便可以將其用修飾性的英語詳細說明，這個寫好的「文件」我們可以稱之爲假想碼（PSEUDOCODE）。意思是你所使用的英文文詞和電腦所認識的命令有些許相似的原故，因此若果以我們的程式爲例可以寫成如圖1的形式。

具體而言，這就可以寫成命令檔如列表1所示，這就是程式設計技巧中所謂TOP－DOWN，STEP－WISE REFI－NEMENT，從頂端開始，然後逐步分開而後征服。

事實上，它只是一種解決大部份問題較顯著的方法而已。首先我們將整個問題做個描述，試著去定義它是甚麽及它不是甚麽。然後再慢慢一步步地更詳細一點，再解決容易解決的詳細的部份，將較複雜的問題放在一邊稍後再解決（或將其分成幾個部份去解決）。

在我們例子裏的這些步驟，我們並沒有作任何報表之類的命令，因爲這對程式的編寫技巧沒有甚麽關係。

而事實上，我們最好不要爲程式的細節煩惱，只要我們能集中精神於整個問題的解決之上。我們可以在測　整個解答和清除這些程序之後再回轉頭來解決。

現在我要結束本文了，各位此時最好是執行一下我們所建的命令檔，究竟在這個簡單程序之中你是否已完全明白並掌握到，若果你不能作肯定的回答最好你重看本文並多作實踐，我可以告訴你必定會成功的！■

程式保護法

· 張健 ·

閣下辛辛苦苦的傑作，一下子就被人抄去眞是可惜。程式的保護形式十分多，筆者也曾使用過LOCK—IT—UP，APPLE GUARD 等軟件作程式的保護，還試過搬移VTOC，但如果對軟件用法和DOS 不太懂之朋友，往往花很長時間，也得不出結果。以軟件和搬移VTOC 的方法，用的時間也不少，所以筆者便提供這短短的程式爲大家保護以DOS 3.3寫的程式，希望大家喜歡。

程式使用方法

㈠ 首先我們要BOOT起普通DOS

㈡ 然後LOAD入HELLO程式

㈢ 鍵入CALL－151進入MONITOR

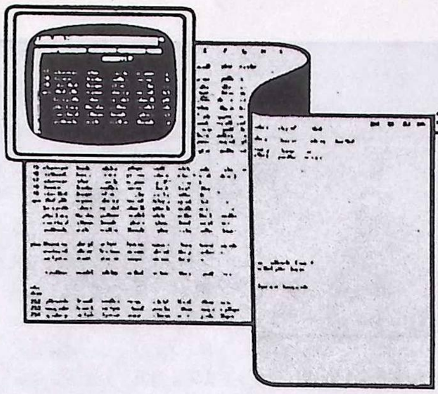
㈣ 鍵入以下程式

```
*B8A3 : 92
*B99B : 92
*B93F : 92
*BCB3 : 92
```

㈤ 按CTRL－C回BASIC

㈥ INIT（FILENAME）如：HELLO

這程式的好處是主磁碟（加了保護的碟）作COPY 時，原磁碟BOOT 起時是一點有保護的迹象也沒有。但那隻BACK－UP，却在BOOT 時會不斷發出軋軋作响的聲音，這樣磁碟機便受苦了，此碟CATALOG也不能。還有一個好處原磁碟在RUN，SAVE 等是絕對不受影响。■



VISICALC

陳耀南

函數@LOOK UP的使用技巧

我們曾經在上期介紹VISICALC 將一些數學函數以圖表來顯示的技巧，在今期的專欄裏我會爲各位介紹一些較高級的VISICALC 函數@LOOKUP 及其使用技巧。

今年3月，中國施行外滙管制，這樣會對貿易造成一定程度的打擊，無巧不成趣，我有一位朋友委託我爲他的公司設計一個程式來①對他公司職員過去十二個月的個別效績作出評估，②改動輸入數據便可作爲業務（或業績）的預測，③分別出最有價值的員工，④計算一年全部的薪示和佣金，⑤順便頒發紅利，⑥作一個總營業額下降的預測，看看打擊有多大。

圖１顯示了我爲他所設計的表，銷售員的名字出現在左邊，月份由一月到十二月出現在頂端，標題則表示了這個表的目的。注意這個表（圖１）的銷售數字是以千元作爲單位的，因爲這樣會簡化資料項。

接著是指示VISICALC 去計算整個月份的銷售額（把一行（COLUMN）加起來）和個人的總銷售額（把一列（ROW）加起來）。然後，祇要很快的按一些鍵，就能夠根據後者的總和建立一個直線圖。

下一步要計算薪水，銷售員每月底薪是一千元，加上5％的銷售佣金，因此在表內建立〝SALARY PLUS COM－MISSION〞行。最後要決定紅利，完成後會如圖１所示。

以下我們要建立一個有12個月銷售資料的表，這個表如圖１所示，程式列表則如圖２所示。這個表的最左邊由A６至A15存放著售貨員的名字，並在表的頂端加入５列（ROW），以留出空間給行的標題和起首，在位置B４到M４，我們鍵入月份的數目（由１到12），以向右靠齊的格式存放（鍵入／FR即可)，之後在位置A４鍵入／TB，固定住兩個方向上的標題。

我們最關心的是計算每個售貨員（SALES）的①全年總銷售額，②總年薪，③紅利。爲了做到這一件事，我們必須學習如何使用一個VISICALC 函數——@LOOKUP 。這個函數與任何目前爲止所看到的函數不同，@LOOKUP 函數需要兩種不同形式（TYPES）的引數（ARGUMENTS）——意即@LOOKUP 需要一個數值和一個範圍，它的格式是：

@LOOKUP（數值，範圍）

這個@LOOKUP 函數最佳用途是處理一些不容忽視的文件，我們需要經常或至少每年總要研究一次的，如稅率表（TAX TABLE）等所鼓舞我們「積極」的文件。它的格式（稅率表）是我們熟悉的；有一行收入在左邊，和一行稅金在右邊。你使用稅賦表由收入行一直看下去，直到找到你自己的收入介於中間的兩項，然後移動你的手指到右邊的行，找出你欠多少稅金。

@LOOKUP 函數能執行這樣的搜尋工作，但是它的運算不限制在稅率表，你可以很生動地使用這個函數來獲得到它的好處，方法很簡單，你只需在表上某處設一相對應的資料行（COLUMN）或列（ROW），放入你的對照表，例如圖A的紅利表（以千元作單位（M行 及 N行）

	M	N
21	BONUS	TABLE
22	------	------
23	0	0.25
24	50	1
25	75	1.25
26	100	2.25
27	125	3.0
28	150	3.5

這個表我們輸入到表中的M和 N行，並放在銷售資料的下方。它的運算方法如下：

若某一個售貨員的年銷售額，設＄85,000 元，則我們由行M看下去，直到＄85,000（或85，因爲上表的數字以＄1000元爲單位）落在兩項中間，當我們找到那裏，我們然後橫過到行N，到行M兩限制項中較少項的那一列，找出紅利。對於一年＄85,000的銷售額，紅利是＄1,500元（因爲85落在75和100中間）。

函數@ LOOK UP 的使用技巧

	1	2	3	4	5	6	7	8	9
john	11.18	6.35	8.13	3.56	2.54	.76	1.52	2.03	1.78
smith	7.62	7.11	5.59	3.3	.25	0	0	0	.51
mary	11.18	7.62	6.35	4.06	1.02	.25	0	0	.51
brown	1.52	1.78	3.05	4.83	6.86	4.83	4.57	3.3	2.79
lisa	4.83	5.33	7.62	9.91	9.91	11.18	9.4	7.37	7.37
marlow	11.43	11.68	13.97	10.67	10.67	11.94	17.02	13.46	14.22
harper	6.6	5.59	7.87	8.89	8.89	8.38	3.89	7.87	7.11
fleming	5.59	5.08	5.33	9.3	15.49	22.86	17.53	17.24	22.1
nash	6.6	6.35	8.38	7.37	9.55	8.89	10.41	12.09	7.87
white	6.86	7.37	9.4	8.38	8.89	7.62	9.4	10.16	8.38
**ave.									

要找出這個值，使用@LOOKUP 函數時，我們可以寫出 @LOOKUP（85，M23…M28）第一個引數，85是我們要尋找的值。第 2 個引數，M23…M28，是我們所要尋找的數值的範圍。若範圍是一行，如這一個例子，則@LOO—KUP 從緊鄰的右邊一行找到對應值傳回，如紅利尋取表上所顯示的。(若範圍是一列，則數值由該列的下一行得到。)

在我們能使用紅利表之前，我們要計算出每個售貨員一年內的總銷售額，然後我們才到尋取表上搜尋對應的紅利值，於是我們就把這些資料放在行N：在程式列表中你可以看到這些以下所鍵入的內容

>N 4／FR TOTALS
>N 5／FR "====
>N 6@SUM（B6.M6）
／R
〔RETURN〕
N7.N15〔RETURN〕
RR

現在，我們要使用@LOOKUP 函數，從位置M21開始，將紅利顯示在行O，在行O 4開始我們所鍵入的內容是：

>O 4／FR BONUS
>O 5／FR "====

在位置O 6，LOOKUP公式是

@LOOKUP（N6，M23…M28），它顯示（指示）VISICALC 由N6 取出銷售總額；到紅利尋取表上M 23 到M28的範圍內找到它的位置，並由N 23到N 28的項中傳回對應的紅利值。（換句話說，VISICALC 在行M尋找指定的值，然後由它的右邊緊鄰的行，即行N，傳回一個值。）當你輸入公式之後，O 6所顯示的值是 1，因此對年銷售額$ 58,680 元的售貨員得到的紅利是$ 1000元。

現在你可以複製公式（複製尋取公式）去找出所有售貨員的紅利，複製的唯一技巧部份是我們需要指定數值參考那些是相對的和那些是常數。很明顯的N 6 這個總銷售數字是相對的。因此我們希望使用不同的數額來計算各人的紅利，但數值參考的範圍，M23…M28是常數。對每個人我們都使用相同的紅利表。這裏是複製命令的使用技術：

首先將CURSOR移到O 6，再鍵入／R〔RETURN〕
O 7. O 15
RNN

要完成這個表，我們還應加上一行總薪水。每年底薪是$12,000元，和佣金是 5 %，我們將這些資料放上紅利表的上方，分別在位置N18和N19：

```
        M       N
17   =====   =====
18   BASE       12
19   COMM.    0.05
20   ====     ====
```

由於總薪水的公式是：底薪＋紅利＋（佣金比率×總銷售額），於是我們將輸入與這個 對 等 的運算式到位置P

```
          annual sales      salary
          jan    to         dec
==================================================================
 10       11       12    totals     bonus  tot sal    best  worst  <ave.
   ===      ===      ===        ===     ===          ====   ====   ====
.81     7.62     9.4      58.68       1  15.934     11.18    .76 **
.76     5.08     5.59     35.81     .25  14.0405     7.62      0 **
.54     5.84    10.16     49.53     .25  14.7265    11.18      0 **
.79     2.03     1.02     39.37     .25  14.2185     6.86   1.02 **
.11     6.35     5.08     91.46     1.5  18.073     11.18   4.83
.84     9.91    12.95    143.76       3  22.188     17.02   5.84
6.6     7.11     6.1      84.9      1.5  17.745      8.89   3.89
.83     6.86     4.06    152.27     3.5  23.1135    22.86   4.06
.86     7.37     7.62     99.36     1.5  18.468     12.09   6.35
7.37    9.65     8.89    102.37    2.25  19.3685    10.16   6.86

        ================                        11.904  3.361
        base             12                 ave.tot 85.751
        comm.           .05
        ================
        bonus  table
        ----------------
               0        .25
              50          1
              75        1.5
             100       2.25
             125          3
             150        3.5
```

6：　＋O６＋N 18＋（N 19＊N６）

然後複製公式到所有的售貨員，按入如下：

／R

〔RETURN〕

PT.P15 〔RETURN〕

RNNR

最後，我們可以加入一個標題到表的頂端我們一開始就保留下來的列。如圖１所示。

我們現在要最後一次回到我們的售貨員表。而我們假定的情況是這樣的：你需要設定某種客觀的標準來決定那些售貨員做不好。當然，你也許會在決定採取任何行動之前，加上一些主觀的判斷到這項準則之上。但至少客觀的準則可以在 VISICALC 的表上算出來，這個準則將基於三項可以由表內銷售數據計算出來的平均數：①每個售貨員平均整年銷售額，②所有售貨員的最佳單月銷售數字的平均值（即找出各售貨員在十二個月當中，最佳的月份銷售數字，把所有這些數字對十個售貨員加以平均，③售貨員最差單月銷售數字的平均值。

任何售貨員的記錄無法配合或超過至少其中一項類別的平均值之時，就要付諸檢定和重新考核。

第一項平均值可以直接由全年銷售數字——行N而計算出來，那是我們早先已經放在表內的，我們把這個數字放到位置R18，R18的公式是：

@AUERAGE（N6…N15)，你也可能會希望用一個標註放在位置Q 18來說明這個值，因此我們可以在Q18中鍵入〝AVE. TOT＝ 來代表平均總數。要計算其它兩個平均值，我們需要設兩行作各售貨員的最佳和最差單月銷售記錄之用。我們使用@MAX和@MIN 函數來產生那些行。最佳單月銷售數字在行Q，而最差的在行R，這裏是設這兩行的指令的順序：

＞Q４／FR　BEST

＞Q５／FR　〝＝＝＝＝

＞Q６@MAX（B6.M６）

＞R４／FR　WORST

＞R５／FR　〝＝＝＝＝

＞R６@MIN（B6.M６）

＞Q６／R

.R６〔RETURN〕

Q7.Q15〔RETURN〕

RRRR

下一個步驟是我們要把最佳和最差行的平均值放在Q 17和R 17。公式是：

＞Q 17@AVERAGE（Q６…Q 15)

＞R 17@AVERAGE（R６…R 15)

爲了代表這兩行，你可以在位置A 17加上一個註標如〝＊＊AVE.〞，如圖１所示。

現在我們已經計算好三個平均值，作爲售貨員評估公式的基礎。現在把CURSOR 置於S4，鍵入行的標題在S４上，〝＜AVE.〞(小於平均值的註標），並在位置S5 加上UN—DERLINE，我們將公式本身收到位置S６。現在我們要

函數@ LOOK UP 的使用技巧

考慮的是，那一個是最好的辦法來指示售貨員的記錄低於平均值呢？一個好方法是包含那些記錄的列中放一個某種形式的“旗標”FLAG，我們已經知道／F＊是一個很方便的格式命令，它會把數字變成星號；在上期專欄我已經介紹過其使用方法。

現在把CURSOR 置於位置S6，鍵入／F＊。我們可以用@AND 這個函數直接表達我們的公式，我們先爲第一個售貨員寫好公式，然後複製到其它的售貨員去。

若果我們要審查一個售貨員，應該以下面3個條件作爲準則：①該售貨員的全年銷售額必須小於平均年銷售額即N6＜R 18。②他的最佳月銷售額必須小於平均最佳月銷售額，即Q6＜Q17 。③他的最差月銷售額必須小於平均最差月銷售額，即R6＜R17 。因此我們將這3個條件合併在S 6裏，即@AND（N6＜R18，Q6＜Q17，R6＜R17），若果這個@AND 函數對某個售貨員傳回眞值，則銷售記錄低於所有三項類別的平均值，而且我們希望設一個旗號給它。此處是我們要如何產生這個旗號的方法，我們用@IF函數，使得眞值時放一個2到位置裏，而假值時放一個0。在／F＊的格式下，2會產生兩個星號，而0則保持空白，於是這裏是我們鍵入到S6 的公式：

@IF（@AND（N6＜R18，Q6＜Q17，R6＜R17）2，0）

而這個@IF函數包含三個所要的引數：邏輯值來自@AND函數，對應眞值的是2，對應假值的是0，兩個星號會出現在位置S6，指示第一個售貨員的銷售記錄低於平均值。

最後，我們要複製這公式到所有的售貨員，目的範圍是S7到S15 。在複製的第三階段，你需要仔細地考慮數值參考；其中三個是相對的，個別的銷售記錄，另外三個在整個複製當中保持常數（即平均值）：

RNRNRN

最後的結果就如圖1所示，我在這裏要強調的是各位朋友可以將這個例子（程式）作爲借鏡去做一些你需要的事，因爲VISICALC的使用範疇是很廣泛的。■

```
>N28:3.5
>M28:150
>N27:3
>M27:125
>N26:2.25
>M26:100
>N25:1.5
>M25:75
>N24:1
>M24:50
>N23:.25
>M23:0
>N22:/--
>M22:/--
>N21:"table
>M21:"bonus
>N20:/-=
>M20:/-=
>N19:.05
>M19:"comm.
>R18:@AVERAGE(N6...N15)
>Q18:"ave.tot=
>N18:12
>M18:"base
>R17:@AVERAGE(R6...R15)
>Q17:@AVERAGE(Q6...Q15)
>N17:/-=
>M17:/-=
>A17:"**ave.
>S15:/F*@IF(@AND(N15<R18,Q15<Q17,R15<R17),2,0)
>R15:@MIN(B15...M15)
>Q15:@MAX(B15...M15)
>P15:+O15+N18+(N19*N15)
>O15:@LOOKUP(N15,M23...M28)
>N15:@SUM(B15...M15)
>M15:8.89
>L15:9.65
>K15:7.37
>J15:8.38
>I15:10.16
>H15:9.4
>G15:7.62
>F15:8.89
>E15:8.38
>D15:9.4
>C15:7.37
>B15:6.86
>A15:"white
>S14:/F*@IF(@AND(N14<R18,Q14<Q17,R14<R17),2,0)
>R14:@MIN(B14...M14)
>Q14:@MAX(B14...M14)
>P14:+O14+N18+(N19*N14)
>O14:@LOOKUP(N14,M23...M28)
>N14:@SUM(B14...M14)
>M14:7.62
>L14:7.37
>K14:6.86
>J14:7.87
>I14:12.09
>H14:10.41
>G14:8.89
>F14:9.55
>E14:7.37
>D14:8.38
>C14:6.35
>B14:6.6
>A14:"nash
>S13:/F*@IF(@AND(N13<R18,Q13<Q17,R13<R17),2,0)
>R13:@MIN(B13...M13)
>Q13:@MAX(B13...M13)
>P13:+O13+N18+(N19*N13)
>O13:@LOOKUP(N13,M23...M28)
>N13:@SUM(B13...M13)
>M13:4.06
>L13:6.86
>K13:20.83
>J13:22.1
>I13:17.24
>H13:17.53
>G13:22.86
>F13:15.49
>E13:9.3
>D13:5.33
>C13:5.08
>B13:5.59
>A13:"fleming
```

函數@ LOOK UP 的使用技巧

```
>S12:/F*@IF(@AND(N12<R18,Q12<Q17,R12<R17),2,0)
>R12:@MIN(B12...M12)
>Q12:@MAX(B12...M12)
>P12:+O12+N18+(N19*N12)
>O12:@LOOKUP(N12,M23...M28)
>N12:@SUM(B12...M12)
>M12:6.1
>L12:7.11
>K12:6.6
>J12:7.11
>I12:7.87
>H12:3.89
>G12:8.38
>F12:8.89
>E12:8.89
>D12:7.87
>C12:5.59
>B12:6.6
>A12:"harper
>S11:/F*@IF(@AND(N11<R18,Q11<Q17,R11<R17),2,0)
>R11:@MIN(B11...M11)
>Q11:@MAX(B11...M11)
>P11:+O11+N18+(N19*N11)
>O11:@LOOKUP(N11,M23...M28)
>N11:@SUM(B11...M11)
>M11:12.95
>L11:9.91
>K11:5.84
>J11:14.22
>I11:13.46
>H11:17.02
>G11:11.94
>F11:10.67
>E11:10.67
>D11:13.97
>C11:11.68
>B11:11.43
>A11:"marlow
>S10:/F*@IF(@AND(N10<R18,Q10<Q17,R10<R17),2,0)
>R10:@MIN(B10...M10)
>Q10:@MAX(B10...M10)
>P10:+O10+N18+(N19*N10)
>O10:@LOOKUP(N10,M23...M28)
>N10:@SUM(B10...M10)
>M10:5.08
>L10:6.35
>K10:7.11
>J10:7.37
>I10:7.37
>H10:9.4
>G10:11.18
>F10:9.91
>E10:9.91
>D10:7.62
>C10:5.33
>B10:4.83
>A10:"lisa
>S9:/F*@IF(@AND(N9<R18,Q9<Q17,R9<R17),2,0)
>R9:@MIN(B9...M9)
>Q9:@MAX(B9...M9)
>P9:+O9+N18+(N19*N9)
>O9:@LOOKUP(N9,M23...M28)
>N9:@SUM(B9...M9)
>M9:1.02
>L9:2.03
>K9:2.79
>J9:2.79
>I9:3.3
>H9:4.57
>G9:4.83
>F9:6.86
>E9:4.83
>D9:3.05
>C9:1.78
>B9:1.52
>A9:"brown
>S8:/F*@IF(@AND(N8<R18,Q8<Q17,R8<R17),2,0)
>R8:@MIN(B8...M8)
>Q8:@MAX(B8...M8)
>P8:+O8+N18+(N19*N8)
>O8:@LOOKUP(N8,M23...M28)
>N8:@SUM(B8...M8)
>M8:10.16
>L8:5.84
>K8:2.54
>J8:.51
>I8:0
>H8:0
>G8:.25
>F8:1.02
>E8:4.06
>D8:6.35
>C8:7.62
>B8:11.18
>A8:"mary
>S7:/F*@IF(@AND(N7<R18,Q7<Q17,R7<R17),2,0)
>R7:@MIN(B7...M7)
>Q7:@MAX(B7...M7)
>P7:+O7+N18+(N19*N7)
>O7:@LOOKUP(N7,M23...M28)
>N7:@SUM(B7...M7)
>M7:5.59
>L7:5.08
>K7:.76
>J7:.51
>I7:0
>H7:0
>G7:0
>F7:.25
>E7:3.3
>D7:5.59
>C7:7.11
>B7:7.62
>A7:"smith
>S6:/F*@IF(@AND(N6<R18,Q6<Q17,R6<R17),2,0)
>R6:@MIN(B6...M6)
>Q6:@MAX(B6...M6)
>P6:+O6+N18+(N19*N6)
>O6:@LOOKUP(N6,M23...M28)
>N6:@SUM(B6...M6)
>M6:9.4
>L6:7.62
>K6:3.81
>J6:1.78
>I6:2.03
>H6:1.52
>G6:.76
>F6:2.54
>E6:3.56
>D6:8.13
>C6:6.35
>B6:11.18
>A6:"john
>S5:/FR"====
>R5:/FR"====
>Q5:/FR"====
```

```
>P5:"===
>O5:"===
>N5:"===
>M5:"===
>L5:"===
>K5:"===
>J5:"===
>I5:"===
>H5:"===
>G5:"===
>F5:"===
>E5:"===
>D5:"===
>C5:"===
>B5:"===
>S4:/FR"<ave.
>R4:/FR"worst
>Q4:/FR"best
>P4:/FR"tot sal
>O4:/FR"bonus
>N4:/FR"totals
>M4:+L4+1
>L4:+K4+1
>K4:+J4+1
>J4:+I4+1
>I4:+H4+1
>H4:+G4+1
>G4:+F4+1
>F4:+E4+1
>E4:+D4+1
>D4:+C4+1
>C4:+B4+1
>B4:1
>S3:/-=
>R3:/-=
>Q3:/-=
>P3:/-=
>O3:/-=
>N3:/-=
>M3:/-=
>L3:/-=
>K3:/-=
>J3:/-=
>I3:/-=
>H3:/-=
>G3:/-=
>F3:/-=
>E3:/-=
>D3:/-=
>C3:/-=
>B3:/-=
>A3:/-=
>O2:"dec
>N2:"to
>M2:"jan
>O1:"salary
>N1:"sales
>M1:"annual
/W1
/GOC
/GRM
/GC7
>N1:/L9
>O1:/L9
>P1:/L9
/X>A1:>A1:/TV
>A1:/TH
/X!/X>A1:>A1:
```

會發聲的鍵盤

（改良篇）

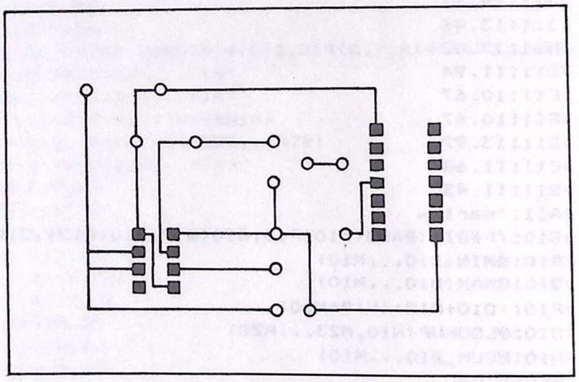
胡永達

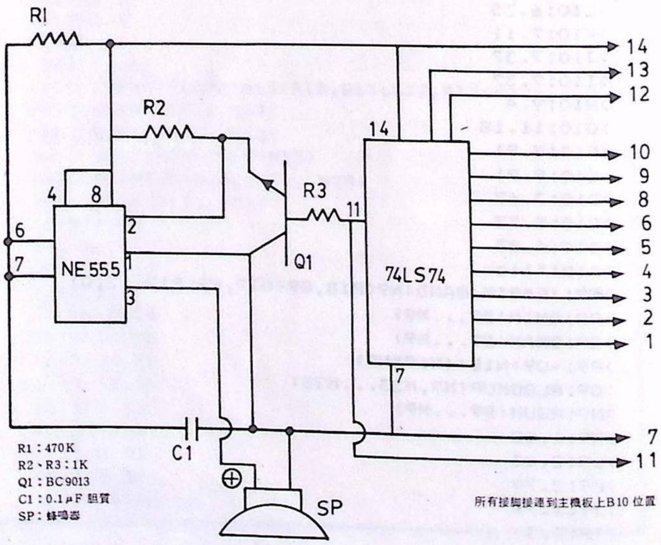
本人在第十三期之中讀到「亞崇」的一篇：發聲的鍵盤，而感到興趣，爲方便讀者們製作，將「亞崇」所設計的雛型加以改良，但先請「亞崇」見諒。

本人的改良品綫路不但非常簡單，而且其聲音與IBM相同，而最大的優點是零件極少，安裝容易及花費極少。製作如下：

我們只需要之零件包括：一枚NE555 、一粒電容、三粒電阻、一粒BC9013 及一個蜂鳴器。其好處是利用蜂鳴器可以將聲音顯得近似IBM 一些。本製品是利用JUMPER來做連接，所以安裝是非常容易。本製品主零件之一74LS74更加可以在AP Ⅱ拆除的，所以花費極少。本製品之連接在AP Ⅱ之B10 之位即74LS 74 。希望讀者歡喜。

大家只要依照附圖來裝嵌即可。■





R1：470K
R2、R3：1K
Q1：BC9013
C1：0.1μF 鉭質
SP：蜂鳴器

所有接腳接連到主機板上B10 位置

機械語言入門

吳博士

機械語言怎樣控制磁碟機馬達活動（下）

在上期我曾經討論過如何以機械語言來直接控制磁碟機的馬達及步進馬達的方法，而且亦列出過一個以BASIC寫成的程式作爲範例，說明採用BASIC的POKE 指令亦可達到和機械語言相同的效果，若果你沒有看過上一期專欄的話，請你在繼續本文之前先參看一回，就算閣下已經看過，亦請再一次翻閱，使可以加強你的記憶！

上期所看到的BASIC程式是一個最簡單移動磁臂（磁頭）的範例，因爲磁臂是由步進馬達（STEPPER MOTOR）的相對相位（PHASE）來控制的原故，我們有必要去清楚了解步進馬達的相位狀態，因爲這個原故，我們要先研究DOS。

DOS佔用一些TEXT 螢幕的記憶位址，不過這些位址無論放入甚麽內容都不會影响正常的TEXT 螢幕輸出，因爲這些位置是在40行×24列的螢幕範圍以外，故此螢幕上是不會顯示出來的，而我們若果知道這些DOS 佔用位址上的資料內容，便可以由此推論出步進馬達的相位，亦知道磁臂的相對位置。

要移動磁碟臂，應該採用以下列出的步驟：

①啓動磁碟機馬達。
②通電流往步進馬達的磁極。
③延遲一段時間使磁臂可被校正到正確位置上。
④若要跳往另一條磁軌的話，則再次執行以上的第二及第三步驟。

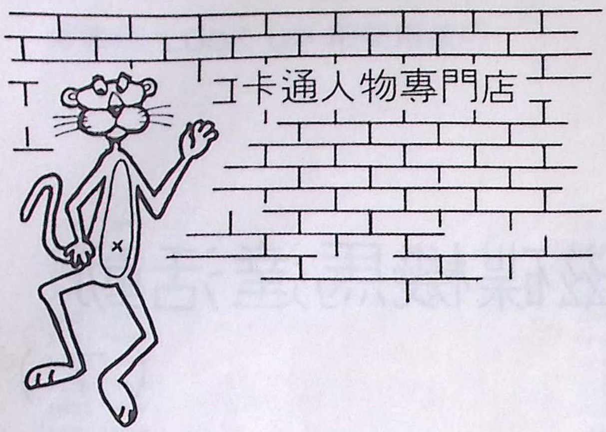
以上的四個步驟最重要的地方是時間延遲的技術，DOS在＄BA11至＄BA28 位址存有一些數值來給予它的SEEK一ABS副程式（位於＄B9A0）產生延遲時間。若果要將磁頭（磁臂）的移動速度提高，則可以將＄BA11至＄BA28表中的內容降低，而DOS延遲最長時間是第一個表上的數值，在此之後所延遲的時間便會減低，因此可以改動這些參數來使DOS的磁頭移動加快一些。

各位如果執行過上期所刊出的BASIC程式時便會知道該程式可以很有效地移動磁頭，因爲BASIC 的組譯器（BASIC INTERPRETER）的執行速度十分緩慢，因此我們在BASIC 程式之上是無需加入延遲網路入內。以下是該程式的功能解說：

- 行號10是用來清除螢幕，將磁碟機所在的SLOT存入SL，並將磁碟機號碼存入到DR之中。
- 行號20將現時步進馬達的相位存入CP，將目的相位存入到DP之上，使程式知道磁頭將要移往那一個相位去。
- 行號30將指定磁碟機的控制暫存器的開始位址存入AD之中，再啓動磁碟機的馬達，即將＄C0N9的數值存入累加器之上，並視乎DR的數值載入＄C0NA或＄C0NB 的數值到累加器來通電流啓動磁碟機馬達。
- 行號40是用來檢查現時步進馬達相位與目的相位數值是否相同，若果相同的話則跳往110行終止程式。
- 行號50用來決定磁頭要移動的方向，若果目的相位比現時相位數值較大時，則磁頭便要向前移到較高磁軌，此責任交由第80行執行。若果目的相位較現時相位少的話則交由第60行執行。
- 行號60是設立一個迴圈來計算向後移動磁頭的次數。
- 行號70是用來存取步進馬達的相位磁極的數值，以開啓四個相位中的其中一個磁極，再檢查迴圈是否完成，若是的話則跳到第110行終止程式。
- 行號80是用來設立一個迴圈來向前計算兩次一跳到達目的相位。
- 行號90是用來存入下一個要啓動的相位磁極的暫存器內容。
- 行號100是用來存入在90行中開啓過的暫存器一個關閉它的暫存器內容。
- 行號110是用來關閉磁碟機馬達並中止程式。

若果我們對這個程式加以改動的話，便可以用在大型程式之上作爲經由DOS而直接控制磁碟機的副程式。但我們要使用它在四份一磁軌之上工作之時，以我的意見認爲是不適宜的，以DOS來說，它只存取磁碟的偶數相位，這是因爲步進馬達精確度是不能用於四份之一磁軌上的，以避免映像干涉。不過我在這裏仍要解釋一下四份一磁軌是些甚麽。

四份之一磁軌的存取資料方式是幾乎在同一時間之內開啓步進馬達中的兩個相鄰相位，由於磁頭上的永久磁石會被兩個相位的磁極牽引，使磁軸被移到它們中間位置，換句話說是在兩個相位的中間。因此若果資料要寫入第0相位的話便會令第1相位上的資料被覆蓋，閣下不妨想一下究竟在第零相位與第0.5相位之間存放資料會引致多少磁碟上資料被覆蓋掉？我可以告訴你，這樣會覆蓋掉幾乎所有磁軌上的資料（0一70相位上的資料），既然如此，四份之一磁軌的資料存取方法便無價値去深入討論了。■







擁有印字機的讀者由今期開始更加可充份發揮它的用途。電腦時代經常刊登多類不同的程式，務求能配合發揮讀者們所擁有的各類週邊設備，例如128K RAM CARD，MOCKING BOARD，印字機等。

這個新開闢的專欄是供擁有印字機的讀者每月編印出一些有趣的著名公仔人物與及中國的風景（例如北京天壇）。假如大家再利用電腦時代第一期的〔電腦時代月曆〕使用，更可成爲精美的掛牆月曆，大家不妨留意。

今期要爲大家介紹的公仔是著名的電視卡通片集藍精靈的主角：藍精靈。各位可依照本文列表打入電腦即可，之後用SAVE CARTOON STAR NO.1存入檔案，或在同期出版的程式磁碟上找到同名的檔案。

大家打入程式時可留意在末段的一DATA 句子中，筆者加插了自動檢錯CHECK SUM，假如打入的DATA有錯，RUN一次後即會發覺CHECK SUM是會有所不同。■

```
1  REM  ***************************
5  REM  SMURF'S FATHER PRINTING
6  REM  ***************************
10  CLEAR : HOME :L = 1010
11  PR# 1: PRINT  CHR$ (24)
12  PRINT  CHR$ (27)"O"; CHR$ (27
   )"E
20 H$ = "WRITTEN BY HO SANG HOU"
21 A(0) =  ASC (" ")
22 A(1) =  ASC ("#")
30 A =  LEN (H$)
50  READ X: IF X < 0 THEN 100
55 A = (A + 1) * (A < 21)
56 D =  ASC ( MID$ (H$,A + 1,1)) -
    65: IF D < 0 THEN 55
60  IF X > 60 THEN 200
70 C = C +  ABS (X)
71 W = (W + 1) * (W < 1)
72  FOR H = 1 TO X - D
73  PRINT  CHR$ (A(W));: NEXT
80  GOTO 50
100  IF X =  - 999 THEN 950
105  PRINT
106  PRINT  TAB(  - X);:W = 1
107 W = 0
110 C =  - X: GOTO 50
200  IF X < > C + D THEN 900
201 W = 0
210 L = L + 10: GOTO 50
900  PRINT : PRINT "CHECK SUM ERR
    OR IN LINE ";L
910  PRINT : END
950  PRINT : PRINT : PRINT : PR#
    0
960  PRINT : PRINT "THAT IS O.K."
    : PRINT
1010  DATA  -27,34,078
1020  DATA  -24,25,068
1030  DATA  -21,25,15,18,080
1040  DATA  -19,28,25,17,107
1050  DATA  -17,04,34,08,070
1060  DATA  -16,17,42,24,116
1070  DATA  -14,11,42,21,092
1080  DATA  -13,16,24,26,086
1090  DATA  -12,16,41,03,085
1100  DATA  -11,08,29,17,085
1110  DATA  -10,24,37,12,102
1120  DATA  -09,21,23,17,071
1130  DATA  -08,26,21,22,095
1140  DATA  -08,01,27,10,10,16,09
    2
1150  DATA  -07,24,38,10,098
1160  DATA  -07,20,22,29,079
1170  DATA  -06,26,22,17,33,04,12
    1
1180  DATA  -06,07,21,16,41,25,13
    3
1190  DATA  -05,10,32,20,29,15,11
    2
1200  DATA  -05,26,26,15,23,10,18
    ,08,138
1210  DATA  -05,16,38,24,25,10,21
    ,24,06,19,189
```

```
1220 DATA  -04,27,25,15,34,04,19
 ,07,142
1230 DATA  -04,16,38,24,26,13,21
 ,20,13,14,190
1240 DATA  -04,26,25,16,20,05,14
 ,07,12,17,30,23,35,12,265
1250 DATA  -04,21,21,15,03,25,12
 ,17,21,03,24,09,21,16,24,24,
 277
1260 DATA  -04,10,35,20,05,14,02
 ,25,19,17,29,01,15,07,18,16,
 28,23,305
1270 DATA  -04,10,34,21,07,14,09
 ,26,10,15,30,02,14,07,17,15,
 30,23,305
1280 DATA  -04,10,33,21,08,14,08
 ,30,20,15,19,01,14,07,15,15,
 31,23,305
1290 DATA  -04,10,33,20,10,14,08
 ,27,21,15,20,01,14,07,14,15,
 31,23,304
1300 DATA  -04,11,32,21,10,17,05
 ,25,21,15,22,02,19,07,14,15,
 23,23,303
1310 DATA  -05,10,31,20,06,16,08
 ,28,20,16,23,02,19,07,15,17,
 21,23,304
1320 DATA  -05,10,31,20,09,15,04
 ,33,15,16,22,01,14,08,13,15,
 29,23,20,09,331
1330 DATA  -06,20,16,14,08,27,16
 ,16,20,04,16,07,09,15,22,23,
 23,09,22,25,09,14,342
1340 DATA  -06,26,17,15,26,01,35
 ,07,08,15,22,23,23,09,29,20,
 08,14,305
1350 DATA  -06,26,17,15,26,01,34
 ,07,12,15,26,23,27,09,23,20,
 291
1360 DATA  -07,15,10,25,08,17,21
 ,01,22,08,13,16,28,23,23,09,
 30,20,07,14,318
1370 DATA  -07,26,12,18,19,03,17
 ,07,16,16,25,24,25,09,26,20,
 14,14,04,25,334
1380 DATA  -08,16,20,02,18,07,14
 ,15,23,23,18,09,22,26,12,14,
 09,25,12,16,20,01,14,09,360
1390 DATA  -09,17,26,23,25,09,22
 ,20,05,14,04,30,15,15,26,01,
 20,11,10,15,337
1400 DATA  -10,23,23,09,29,20,06
 ,14,02,25,10,19,26,01,21,08,
 14,15,26,23,341
1410 DATA  -09,09,25,20,16,14,09
 ,27,15,16,23,03,21,08,13,15,
 263
1420 DATA  -09,23,19,12,56,22,14
 ,14,02,25,12,15,241
1430 DATA  -09,01,14,07,10,15,51
 ,24,19,10,30,21,07,18,237
1440 DATA  -09,25,11,15,49,01,14
 ,09,19,15,26,23,233
1450 DATA  -10,09,22,20,33,15,24
 ,25,165
1460 DATA  -10,15,22,01,36,07,09
 ,17,33,25,23,10,227
1470 DATA  -11,21,06,14,23,26,08
 ,15,20,02,21,10,10,16,22,24,
 266
1480 DATA  -13,12,38,21,05,14,02
 ,25,16,17,27,02,205
1490 DATA  -16,08,22,17,22,23,19
 ,09,20,20,08,16,201
1500 DATA  -17,27,13,15,20,04,22
 ,08,09,16,171
1510 DATA  -19,31,18,09,32,22,13
 5
1520 DATA  -23,17,02,25,22,15,12
 2
1530 DATA  -24,02,15,07,22,15,10
 5
1540 DATA  -23,23,37,09,111
1550 DATA  -22,20,11,14,14,25,11
 3
1560 DATA  -20,16,25,01,27,07,10
 3
1570 DATA  -19,15,28,23,33,09,14
 6
1580 DATA  -18,20,12,14,18,25,11
 4
1590 DATA  -11,18,20,01,21,07,25
 ,15,138
1600 DATA  -08,25,21,10,27,20,23
 ,14,149
1610 DATA  -07,25,23,15,38,01,12
 2
1620 DATA  -06,07,23,16,41,23,13
 3
1630 DATA  -05,09,34,22,05,15,20
 ,25,142
1640 DATA  -05,15,21,01,27,07,09
 ,15,38,23,178
1650 DATA  -05,09,20,21,19,14,04
 ,25,24,15,174
1660 DATA  -06,01,31,07,10,16,34
 ,23,145
1670 DATA  -05,09,38,20,09,15,13
 ,25,141
1680 DATA  -05,15,21,03,25,07,15
 ,17,25,26,176
1690 DATA  -04,09,22,20,13,15,04
 ,27,17,19,21,01,185
1700 DATA  -04,07,09,15,29,25,19
 ,09,22,20,20,14,194
1710 DATA  -05,26,16,16,26,01,29
 ,07,133
1720 DATA  -06,15,27,24,18,09,26
 ,20,09,14,11,25,11,21,254
1730 DATA  -05,01,20,07,10,15,26
 ,23,24,10,27,20,07,14,08,26,
 250
```

藍精靈

```
1740  DATA  -05,15,24,01,17,07,11
,16,30,27,19,09,22,20,14,14,
252
1750  DATA  -05,25,11,16,24,04,30
,09,09,15,32,23,220
1760  DATA  -06,12,32,20,21,14,02
,25,20,15,185
1770  DATA  -07,01,29,07,24,15,35
,23,158
1780  DATA  -07,09,36,25,15,14,16
,25,154
1790  DATA  -07,15,35,01,18,09,16
,15,34,23,190
1800  DATA  -07,09,44,22,11,14,15
,25,154
1810  DATA  -07,15,31,01,22,08,10
,15,41,23,190
1820  DATA  -08,09,32,22,07,16,06
,25,27,15,185
1830  DATA  -08,01,29,09,16,15,39
,23,157
1840  DATA  -08,09,37,20,10,17,19
,25,152
1850  DATA  -09,15,36,01,17,07,29
,15,149
1860  DATA  -09,23,36,09,21,20,26
,14,159
1870  DATA  -10,25,25,15,20,01,34
,07,144
1880  DATA  -10,15,38,23,18,09,40
,20,177
1890  DATA  -11,14,18,25,08,15,39
,01,144
1900  DATA  -12,07,23,16,41,23,13
9
1910  DATA  -13,09,33,20,05,14,20
,26,147
1920  DATA  -14,15,31,01,15,07,24
,15,142
1930  DATA  -15,24,27,09,22,20,19
,15,152
1940  DATA  -17,26,13,16,23,01,24
,09,136
1950  DATA  -19,20,28,24,23,11,14
4
1960  DATA  -35,25,064
1970  DATA  -999
```

訂閱

（請用英文填寫，方便電腦處理）　（只適用於本港訂戶）

信封面請註明「訂閱」

本人欲訂閱

□電腦時代一年（十二期）HK$120

□電腦時代＋CompuHouse磁碟一年（十二期）HK$475

由第____期開始

姓名：______________

地址：______________

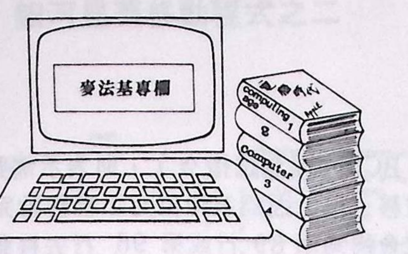
電話：______________

附上劃線支票 No. ______________

支票收款人請填寫 "Computing Age Publisher"

請寄九龍中央郵箱71193號

NO. ______________



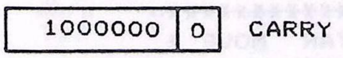
掀開蘋果高解像畫面的神秘面紗

——銀河星羣移動程式之二——

筆者上一期談過的銀河星群移動程式雖然達到星群移動的效果，但程式內並沒有加上測試螢幕上雙單位址部份，所以星群移動時畫面上有叠影出現。原因是倘若我們在畫面上畫上一個位元（BIT），該位元會視乎位址的雙數或單數產生不同顏色，這是 APPLE 高解像畫面一個缺點。

除了顏色問題要解決之外，星群移動的速度亦是一個要改良的地方。很多時我們要獲取星群內的星星分別以不同速度移動，這樣會產生一種立體的感覺。一個位元組（BYTE）的移動已經是位元移動的 7 倍，假如以兩個位元組的速度移動，則星群移動太快，而且缺少了圓滑的感覺。最實際的移動是位元的移動，事實上，所有動畫內的物體如果以 BIT MAP 方法繪圖，都是以位元移動作爲標準的。

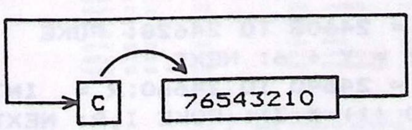
參考下面的圖一，假設圖內的 1 是要移動的位元，我們不斷把它向右移一位，最後它會跑出位元組外，但它並非不存在，而是進入狀態暫存器（PROGRAM STATUS WORD）的進位旗號內（CARRY BIT）。



圖(1)

由於以上這個特性，我們可以知道假如進位旗被設定爲 1 時，則該位元組內的位元已經移動完畢，問題是究竟位元被移動時，該位元組（BYTE）還有些甚麼變化。

一般來說，把位元移動有兩種方法，每種方法



圖(2)

都分爲向左或向右移動，第一種是旋轉的移動（ROL ROTATE ONE BIT LEFT）或（ROR ROTATE ONE BIT RIGHT）。

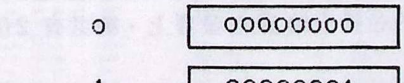
參看圖二，當位元被移向右面一個位置時，它原來的位置已經空出來了，這時最左面空出的位元會被進位旗號的位元代替，而進位旗號的位元則被最右面的位元代替，構成一種旋轉的狀態，位元值是零還是一則視乎該位元組值和進位旗號而定。

這種旋轉的移動方法對於星星移動並不適用，因爲當位元移動時進位位元亦跟着進入位元組內，但我們萬不能讓它進入，如果進位位元是一的話，則會有另外一點星星被繪畫出來。

第二種是移位的移動（ASL ARITHMETIC SHIFT LEFT）或（LSR LOGICAL SHIFT RIGHT）。

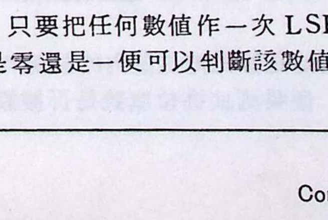
參看圖三，當位元組內的位元向右移動時，最右面的位元會進入進位旗號內，而空出的最左位元則被零加以代替。

這種移位方法是我們需要的一種，尤其是LSR的方法，除了測試位元移動之外，還可以測試該位元組是單數或是雙數。爲甚麽呢？道理很簡單，因



圖(3)

爲所有單數值的最右面位元是 1，而所有雙數值的最右面位元是零，參看圖四：



圖(4)

換言之，只要把任何數值作一次 LSR，然後測試進位旗號是零還是一便可以判斷該數值是雙數還是單數了。

銀河星羣移動程式之二

這裡有些讀者可能不明白，既然我們以 LSR 進行星星移動，但又要以 LSR 測試雙單位置，不是十分混亂嗎？

其實答案十分簡單，進行星星移動是把畫面上位址（ADDRESS）的內裡資料作 LSR 移位，而測試雙單數却是把該位址作 LSR 判斷，兩者是分別進行。LSR 指令的特性是只把累加器（AC-CUMULATOR）的值移位，並不影响任何一個位址或位址內的資料，所以當我們要在畫面上把星星移位時，首先把該位址的資料存放到累加器內然後作一次 LSR ，完成工作後要再把累加器內資料重新放回位址內，畫面上才會有移動的現象，否則不會達到效果。最主要留意的是畫面位址是星星的目前位置，而位址內的資料却是畫面上的不同形狀，只不過星星的形狀就是一個位元而已。

造成這麽麻煩的程序是由於高解像度畫面是由位元構成，位元移動 7 次才移動位元組一次，這是無可避免的事。移動星星只是一個位元，所以由 L-SR 操作不會影响速度，倘若圖形是一個怪物或太空船之類，便要預先造成 7 個不同圖形來達到物體圓滑移動的效果了。

銀河星群內星星分別以三種不同速度向左方移動。最慢是每次移動一個位元，正常速度是每次移動兩個位元，而最快是每次移動三個位元。所以在未呼叫這個程式之前，除了編定一個星星位置表外（參閱上一期），還要編定一個速度表，讓程式根據表內資料把星星配上指定的不同速度。LIS-TING＃1 是一個 BASIC 程式，由 10 至 50 行都和上一期的一樣以亂數把星星的水平和垂直位置放入表內。第 60 行把速度放入另外一個表內，資料分別以零、1 和 2 作爲速度旗號。

現在研究這個銀河群星程式，LISTING＃2 是該程式原始碼。

第 47 行的 DRAWSTAR 是把星星根據 LIS-TING＃1 的位置表繪畫在螢幕上，總共有 20 棵星星。

第 63 行開始是把星星以每次兩個位元的速度加以向左移動。直至第 74 行爲止和上一期談過一樣，使星星到達左面邊緣時再由右面出現。

第 75 行開始把星星在同一位元組內作 LSR 的雙單數判斷，留意 HORIZ 是目前星星的位置，以位元組表示。

第 81 行開始是首先把位置內的位元作移位，每作一次 LSR 便要測試進位旗號是否被設定爲一，如果是的話（BCS 便是測試指令），則表示需要把整個位元組左移一位，因爲這時位元已經移動完畢了，所以程式會跳到第 89 行或第 96 行去移動位元組，視乎雙單數而定。這裡我們發覺總共在第 81 行開始部份作出兩次 LSR 移位，因爲速度是每次移動兩個位元。LSR 次數由速度來決定，但必須每 LSR 一次便要測試進位旗號，千萬不要以爲速度是兩個位元便 LSR 兩次才測試進位旗號。電腦並非超人，它只會在 LSR 之後才會影响進位旗號，而我們不可能知道何時才移位完畢，這是必須的步驟。

因程式內分別建立三個不同速度的星星移動副程式，讓星群移動時根據造好的速度表有所選擇。第 228 行部份便是由速度表內的旗號負責星星的不同速度。

這個程式已經加上詳細的附註說明，相信讀者可以從裡面找到解釋。

讀者要使用這個程式，切記要把上一期刊出的 PIL 一併存放於磁碟上，這是高解像畫面的 LO-OK - UP TABLE 。

各位在輸入程式時留意，LISTING＃1 是一般的 APPLESOFT ，而 LISTING＃2 是利用 MERLIN 編寫的原始碼，懂得 ASSEMBLY 的讀者應該會明白怎樣去輸入，至於初學者可按原始碼左邊的機械碼輸入，然後用 BSAVE STAR M-OVE . OB , A$4000 , L$16B 存入磁碟。

下次再談。■

```
5   REM  **************
6   REM  * STAR  MOVE *
7   REM  * LISTING 1 *
8   REM  **************
10   HGR
20   PRINT  CHR$ (4);"BLOAD STAR M
     OVE.OB"
25   PRINT  CHR$ (4);"BLOAD PIL"
30   FOR I = 24576 TO 24596:R =  INT
     ( RND (1) * 279): POKE I, INT
     (R / 7): NEXT
40 Y = 20
50   FOR I = 24608 TO 24628: POKE
      I,Y:Y = Y + 6: NEXT
60   FOR I = 24640 TO 24660:R =  INT
     ( RND (1) * 3): POKE I,R: NEXT

70   CALL 16398
80   CALL 16384: GOTO 80
```

銀河星羣移動程式之二

```
:ASM
                 ********************************
                 *      LISTING 2               *
                 * STAR MOVE ROUTINE WITH       *
                 *    DIFFERENT SPEEDS          *
                 *  BY PAUL F.K. MAK            *
                 *  20TH MAY,   1985            *
                 *                              *
                 ********************************
                 *
                           ORG  $4000
4000: 4C 67 41             JMP  START
                 *
                 *
                 *
                 *
                 *
                 XBYTE     EQU  $0          ;BYTE POSITION FOR DRAW
                 XBIT      EQU  $1          ;BIT POSTION FOR DRAW
                 VERT      EQU  $2          ;VERT POSTION FOR DRAW
                 HIRESL    EQU  $26         ;LOOK-UP TABLE LO POINTER
                 HIRESH    EQU  $27         ;LOOK-UP TABLE HI POINTER
                 STTEMP    EQU  $FD         ;TEMPORARY VARIABLE
                 SVERT     EQU  $FE
                 HORIZ     EQU  $FF         ;HORIZ POSITION VARIABLE
                 *
                 *
                 *
                 STARBYTE EQU   $6000       ;STAR BYTE POSITION TABLE
                 STARVERT EQU   $6020       ;STAR VERT POSITION TABLE
                 STARSPEEDEQU   $6040       ;STAR SPEED TABLE
                 YVERTH    EQU  $9200       ;LOOK-UP TABLE HIGH
                 YVERTL    EQU  $92C0       ;LOOK-UP TABLE LOW
                 DELAY     EQU  $FCA8       ;DELAY ROUTINE
                 *
                 *
                 *     GET ADDRESS ROUTINE  *
                 *
4003: B9 C0 92   GETADR    LDA  YVERTL,Y    ;LOOK UP LO BYTE OF LINE
4006: 85 26                STA  HIRESL      ;STORE TO LO POINTER
4008: B9 00 92             LDA  YVERTH,Y    ;LOOK UP HI BYTE OF LINE
400B: 85 27                STA  HIRESH      ;STORE TO HI POINTER
400D: 60                   RTS
                 *
                 *
                 **  INITIAL DRAW OF ALL STARS   **
                 *
400E: A2 00      DRAWSTAR LDX   #$00        ;INDEX=0 TO SEARCH VERTICAL TABLE
4010: BC 20 60   DRAWSTAR1LDY   STARVERT,X  ;VERTICAL CO-ORDINATES OF 1ST STAR
4013: 20 03 40             JSR  GETADR      ;FIND LEFT MOST X CO-ORDINATES' AD
DRESS
4016: BC 00 60             LDY  STARBYTE,X  ;HORIZONTAL BYTE VALUEOF 1ST STAR
4019: A9 40                LDA  #$40        ;DRAW ON SCREEN WITH #$01000000
401B: 91 26                STA  (HIRESL),Y  ;STORE VALUE TO PAGE1 GRAPHIC
401D: E8                   INX              ;INCREASE INDEX BY 1
401E: E0 14                CPX  #$14        ;FINISHED DRAWING 20 STARS?
4020: 90 EE                BLT  DRAWSTAR1   ;NO, DRAW NEXT ONE
4022: 60                   RTS              ;YES, DONE ALL STARTS
                 *
                 *
                 *
                 *
                 *     STAR MOVEMENT WITH 2 BITS DISPLACEMENT   *
                 *
4023: A4 FE      STARMOVE LDY   SVERT       ;STAR VERT POSITION
4025: 20 03 40             JSR  GETADR      ;FIND LEFT MOST ADDRESS
4028: A4 FF                LDY  HORIZ       ;ADD HORIZ POSITION
402A: C0 00                CPY  #$00        ;STAR REACHES LEFT?
402C: D0 0F                BNE  STARMOVE1   ;NO, MOVE STARS
402E: A9 00                LDA  #$0         ;YES, ERASE PREVIOUS STAR
4030: 91 26                STA  (HIRESL),Y  ;ON SCREEN
4032: A0 27                LDY  #$27        ;DRAW NEW STAR
4034: 84 FF                STY  HORIZ       ;ON RIGHT MOST POSITION
4036: A9 40                LDA  #$40        ;WITH VALUE #$01000000
4038: 91 26                STA  (HIRESL),Y  ;ON PAGE 1 GRAPHIC
403A: 4C 46 40             JMP  STARMOVE2   ;RETURN
403D: 18         STARMOVE1CLC               ;CLEAR CARRY FLAG FOR TESTING ODD
OR EVEN
403E: A5 FF                LDA  HORIZ       ;EXISTING HORIZONTAL POSITION OF S
TAR
4040: 4A                   LSR              ;SHIFT HORIZ VALUE TO RIGHT
4041: B0 04                BCS  ODD         ;IF CARRY FLAG SET, THEN POSTN IS
ODD
4043: 4C 61 40             JMP  EVEN        ;CARRY FLAG NOT SET, POSTN IS EVEN
4046: 60         STARMOVE2RTS
4047: 18         ODD       CLC              ;CLEAR CARRY FOR DRAW
4048: B1 26                LDA  (HIRESL),Y  ;TAKE EXISTIN ODD HORIZ POSTN OF S
TAR
404A: 4A                   LSR              ;MOVE ONE BIT LEFT
404B: B0 06                BCS  ODDBYTE     ;IF FINISH BIT MOVEMENT, MOVE BYTE
 POSTN
404D: 4A                   LSR              ;NO, MOVE ONE BIT MORE
404E: B0 03                BCS  ODDBYTE     ;TEST AGAIN FOR BIT MOVEMENT FINIS
HED?
4050: 91 26                STA  (HIRESL),Y  ;IF NOT, DRAW NEW STAR POSTIN
4052: 60                   RTS              ;RETURN
4053: A9 00      ODDBYTE   LDA  #$0         ;ODD POSTN OF BYTE MOVEMENT,XDRAW
STAR FIRST
4055: 91 26                STA  (HIRESL),Y  ;CLEAR PREVIOUS POSTN
4057: 88                   DEY              ;MOVE ONE BYTE LEFT
4058: 84 FF                STY  HORIZ       ;TO GET NEW POSITION
405A: A9 40                LDA  #$40        ;DRAW WITH #$01000000
405C: 91 26                STA  (HIRESL),Y  ;ON PAGE1 GRAPHIC
405E: 4C 46 40             JMP  STARMOVE2   ;RETURN
4061: 18         EVEN      CLC              ;CLEAR CARRY FOR DRAW
4062: B1 26                LDA  (HIRESL),Y  ;EXISTING EVEN POSTN OF STAR FOR D
RAW
4064: 4A                   LSR              ;MOVE LEFT ONE BIT
4065: B0 06                BCS  EVENBYTE    ;IF BIT MOVEMENT FINISHED THEN TO
BYTE POSTN
4067: 4A                   LSR              ;NOT FINISHED, MOVE ONE BIT MORE
4068: B0 03                BCS  EVENBYTE    ;BIT MOVEMENT FINISHED?
406A: 91 26                STA  (HIRESL),Y  ;NO, STORE NEW POSITION OF STAR
406C: 60                   RTS              ;TO SCREEN AND RETURN
406D: A9 00      EVENBYTE LDA   #$0         ;EVEN BYTE POSTN, XDRAW STAR FIRST
406F: 91 26                STA  (HIRESL),Y  ;ON SCREEN
4071: 88                   DEY              ;MOVE ONE BYTE LEFT
4072: 84 FF                STY  HORIZ       ;GET NEW HORIZ POSTN
4074: A9 20                LDA  #$20        ;DRAW WITH #$00100000
4076: 91 26                STA  (HIRESL),Y  ;ON SCREEN
4078: 4C 46 40             JMP  STARMOVE2   ;RETURN
                 *
                 *
                 *
                 *     STAR MOVEMENT WITH 1 BYTE DISPLACEMENT   *
                 *     SAME COMMENTS AS ABOVE                   *
                 *
407B: A4 FE      FASTMOVE LDY   SVERT
407D: 20 03 40             JSR  GETADR
4080: A4 FF                LDY  HORIZ
4082: C0 00                CPY  #$00
4084: D0 0F                BNE  FASTMOVE1
4086: A9 00                LDA  #$0
4088: 91 26                STA  (HIRESL),Y
408A: A0 27                LDY  #$27
408C: 84 FF                STY  HORIZ
408E: A9 40                LDA  #$40
4090: 91 26                STA  (HIRESL),Y
4092: 4C 9E 40             JMP  FASTMOVE2
4095: 18         FASTMOVE1CLC
4096: A5 FF                LDA  HORIZ
4098: 4A                   LSR
4099: B0 04                BCS  FASTODD
409B: 4C BE 40             JMP  FASTEVEN
409E: 60         FASTMOVE2RTS
409F: 18         FASTODD  CLC
40A0: B1 26                LDA  (HIRESL),Y
40A2: 4A                   LSR
40A3: B0 0B                BCS  ODDFAST
40A5: 4A                   LSR
40A6: B0 08                BCS  ODDFAST
40A8: 4A                   LSR
40A9: B0 05                BCS  ODDFAST
40AB: 91 26                STA  (HIRESL),Y
40AD: 4C 9E 40             JMP  FASTMOVE2
40B0: A9 00      ODDFAST  LDA   #$0
40B2: 91 26                STA  (HIRESL),Y
40B4: 88                   DEY
40B5: 84 FF                STY  HORIZ
40B7: A9 40                LDA  #$40
40B9: 91 26                STA  (HIRESL),Y
40BB: 4C 9E 40             JMP  FASTMOVE2
40BE: 18         FASTEVEN CLC
40BF: B1 26                LDA  (HIRESL),Y
40C1: 4A                   LSR
40C2: B0 0B                BCS  EVENFAST
40C4: 4A                   LSR
40C5: B0 08                BCS  EVENFAST
40C7: 4A                   LSR
40C8: B0 05                BCS  EVENFAST
40CA: 91 26                STA  (HIRESL),Y
40CC: 4C 9E 40             JMP  FASTMOVE2
40CF: A9 00      EVENFAST LDA   #$0
40D1: 91 26                STA  (HIRESL),Y
40D3: 88                   DEY
40D4: 84 FF                STY  HORIZ
40D6: A9 20                LDA  #$20
40D8: 91 26                STA  (HIRESL),Y
40DA: 4C 9E 40             JMP  FASTMOVE2
                 *
                 *
                 *
                 *
                 *     STAR MOVEMENT WITH 1 BIT DISPLACEMENT    *
                 *     SAME COMMENTS AS ABOVE                   *
                 *
40DD: A4 FE      SLOWMOVE LDY   SVERT
40DF: 20 03 40             JSR  GETADR
40E2: A4 FF                LDY  HORIZ
40E4: C0 00                CPY  #$00
40E6: D0 0F                BNE  SLOWMOVE1
40E8: A9 00                LDA  #$0
40EA: 91 26                STA  (HIRESL),Y
40EC: A0 27                LDY  #$27
40EE: 84 FF                STY  HORIZ
40F0: A9 40                LDA  #$40
40F2: 91 26                STA  (HIRESL),Y
40F4: 4C 00 41             JMP  SLOWMOVE2
40F7: 18         SLOWMOVE1CLC
40F8: A5 FF                LDA  HORIZ
40FA: 4A                   LSR
40FB: B0 04                BCS  ODDSLOW
40FD: 4C 1A 41             JMP  EVENSLOW
4100: 60         SLOWMOVE2RTS
4101: 18         ODDSLOW  CLC
4102: B1 26                LDA  (HIRESL),Y
4104: 4A                   LSR
4105: B0 05                BCS  ODSLOWBYTE
4107: 91 26                STA  (HIRESL),Y
4109: 4C 00 41             JMP  SLOWMOVE2
410C: A9 00      ODSLOWBYTELDA  #$0
410E: 91 26                STA  (HIRESL),Y
4110: 88                   DEY
4111: 84 FF                STY  HORIZ
4113: A9 40                LDA  #$40
4115: 91 26                STA  (HIRESL),Y
4117: 4C 00 41             JMP  SLOWMOVE2
411A: 18         EVENSLOW CLC
411B: B1 26                LDA  (HIRESL),Y
411D: 4A                   LSR
411E: B0 05                BCS  EVSLOWBYTE
4120: 91 26                STA  (HIRESL),Y
4122: 4C 00 41             JMP  SLOWMOVE2
4125: A9 00      EVSLOWBYTELDA  #$0
4127: 91 26                STA  (HIRESL),Y
4129: 88                   DEY
412A: 84 FF                STY  HORIZ
412C: A9 20                LDA  #$20
412E: 91 26                STA  (HIRESL),Y
4130: 4C 00 41             JMP  SLOWMOVE2
                 *
                 *
                 *
4133: A2 00      STAR      LDX  #$00        ;1ST STAR
4135: BD 00 60   STAR1     LDA  STARBYTE,X  ;LOOK UP BYTE POSTN
4138: 85 FF                STA  HORIZ       ;STORE TO VARIABLE
413A: BD 20 60             LDA  STARVERT,X  ;LOOK UP VERT POSTN TABLE
413D: 85 FE                STA  SVERT       ;STORE TO VARIABLE
413F: BD 40 60             LDA  STARSPEED,X ;LOOK UP SPEED TABLE
4142: C9 01                CMP  #$1         ;VALUE=1?
4144: F0 0A                BEQ  S1          ;YES, SLOW MOVE
4146: C9 02                CMP  #$2         ;VALUE=2?
4148: F0 0C                BEQ  S2          ;YES, NOMRAL MOVE
414A: 20 7B 40             JSR  FASTMOVE    ;NO, FAST MOVE
414D: 4C 5C 41             JMP  STAR2
4150: 20 DD 40   S1        JSR  SLOWMOVE
4153: 4C 5C 41             JMP  STAR2
4156: 20 23 40   S2        JSR  STARMOVE
4159: 20 23 40             JSR  STARMOVE
415C: A5 FF      STAR2     LDA  HORIZ       ;NEW HORIZ POSTN OF STAR
415E: 9D 00 60             STA  STARBYTE,X  ;STORE BACK TO BYTE TABLE
4161: E8                   INX              ;NEXT STAR
4162: E0 14                CPX  #$14        ;FINISHED ALL 20 STARS?
4164: 90 CF                BLT  STAR1       ;NO, NEXT ONE
4166: 60                   RTS              ;YES, RETURN
                 *
4167: 20 33 41   START     JSR  STAR
416A: 60                   RTS

--End assembly--

363 bytes

Errors: 0
```

徵求一行程式有獎遊戲

在上月我們收到許多〔一行程式〕的投稿，其中不乏精彩的作品。

這個遊戲旨在鼓勵大家動腦筋，利用最簡短的程式，設計一些有創意的效果。

今次未刊登你的作品的讀者請勿灰心，可能我們會在下一期登出。

各位投稿者請在來稿中加上程式內容說明。以前各期的得獎者請盡快前來讀者服務部領取應得的獎品。

各位讀者寫來程式時，必須用多行的形式寫來（參考本文的各位得獎例子），而且要書寫清楚。投稿請寄來CPO BOX 71193 。入選者可獲得任何一期的程式磁碟一張（請註明要的磁碟）。來稿信封請勿封口。 ■

```
3     HOME
  : A$ = "COMPUTING AGE"
  : PRINT SPC( (40 - LEN (A$)) / 2);A
     $
  : POKE 34,1
  : FOR I = 768 TO 774
  : READ B
  : POKE I,B
  : NEXT I
  : POKE 1010,0
  : POKE 1011,3
  : CALL - 1169
  : DATA 169,1,133,34,76,208,3
```

這個一行程式，能把一般的 DOS 3.3 程式磁碟加以保護，那麼標準的 COPY A 程式，便不能拷貝及 CATALOG 。只要執行本程式後，SAVE 需要保護的 FILE ，而該 FILE 檔案名字用 HELLO，那麼你便有一張加了保護的磁碟。

HO CHI KEUNG 君可取得第十五期程式磁碟一張。

```
1      VTAB 23
   : PRINT "? ";
   : GET A$
   : PRINT A$;":FROM I/F/N",
   : GET B$
   : PRINT "TO I/F/N";
   : GET C$
   : A = ASC (A$) - ( ASC (A$) > 63)
      * 64
   : FOR I = 1024 TO 2039
   : H = PEEK (I) = A + (B$ = "F")
      * 64 + (B$ = "N") * (128 + (A
      < 32) * 64)
   : POKE I,(H = 0) * PEEK (I) + (H
      = 1) * (A + (C$ = "F") * 64
      + (C$ = "N") * (128 + (A < 32)
      * 64))
   : NEXT
   : PRINT "";
```

作　用：本程式是一個畫面（TEXT）處理。它能讓現全的畫面狀態下，將閣下所選擇的字元，由 NORMAL，FIASH，或者 IN－VERSE 轉爲 NORMAL，FIASH 或 INVERSE 來顯示。

方　法："RUN"後，程式會問你所要求的"鍵"，是用 GET A$ 來做的；跟着便問你把這個字元從什麽狀態"FROM I/F/N"轉做什麽狀態"TO I/F/N"只要按下所需的字母便行。

結　果：程式需要 50 秒的時間，才能把整個文字畫面處理。你可以試用空白鍵（SPACE BAR）來看看，結果是非常之有趣的。完後會有CTRL－G的响聲。

輸入法：由於程式是很長，所以"PRINT"專用"？"代替；或把英文句縮短。

何智業可取得第十五期的程式磁碟一張。

本程式當執行後，會在螢幕上方的中央顯示出"COMPUTING AGE"字樣，一如開機時出現的"APPLE]["字樣。之後，無論是寫程式或做其他工作時，這字句都不會消失，就算按"CTRL-RESET"也不能將它除去。只有用"TEXT"才可把它消除！各位可隨意改動 A$ 字串的內容。

劉超毅同學可贏得第十六期程式磁碟一張。

```
10     POKE 47267,146
   : POKE 47515,146
   : POKE 47423,146
   : POKE 48307,146
   : PRINT CHR$ (4);"INIT HELLO"
   : PRINT CHR$ (4);"DELETE HELLO"
   : END
```

本程式是照着應屆會考生的功課"MEMORY TEST"程式而寫的。其中除了沒有用到副程式一點外，其他要求大致也做到了。

開始時用 RND(1) 做出英文字，然後轉到變數 B$ 而再行印出。過了大約半秒就做 HOME 的工作，隨後是由用者鍵入答案。答對就加十分（S），否則不加分。而旗號（K）一定會加十分，因爲這是用來測試是否答對用的。最後 LEVEL 升級，然後再做一次。到程式完結時會印出分數及答案。

最後一提，本程式共有六十個難度，也是說變數 B$ 可以存入六十個字，之後就會有"ERROR"的發生。但人類沒有可能單用腦力記着超過六十個不同的英文字母。而本人最多也是記着三十二個字。（四個人）

黃進光同學請與編輯部聯絡需要那期程式磁碟。

```
1      POKE 34,1
   : IF K = S THEN HOME
   : B$ = B$ + CHR$ ( INT ( RND (1)
      * 26) + 65)
   : VTAB ( INT ( RND (1) * 20) + 2)
   : HTAB ( INT ( RND (1) * 5) + 1)
   : PRINT B$
   : FOR I = 1 TO 302 - L
   : NEXT I
   : HOME
   : INPUT "";A$
   : S = S + (A$ = B$) * 10
   : K = K + 10
   : L = L + 5
   : VTAB 1
   : HTAB 1
   : PRINT "SCORE IS "S
   : HTAB 1
   : PRINT "ANSWER IS "B$
   : GOTO 1
```

E.A. 遊戲軟件保護方法

Alen

前言

ELECTRONIC ART 出品的遊戲種類繁多，而且能保持非常高的水準。不過，它的產品所附加上的保護亦是較其他公司複雜，要解除它的保護必須下一番苦功。本文作者一向以來，對拆碟解保護具有相當的心得，現在由他來撰寫解拆E.A. 產品所累積得來的心得，相信對各讀者日後研究E. A. 產品的破解方法必定有很大的帮助。

「E. A.」——ELECTRONIC ART，這間軟件公司是近期風頭較盛的，而它的產品售價亦相當昂貴，大約在＄40美金左右。但由於它的SOFTWARE有一定的吸引力，所以它的產品銷路亦相當之高。

它的產品可以在香港買到的大致有：AXIS ASSASSIN、HARD HAT MARK、THE LAST GLADIATOR、ONE ON ONE、ARCHON、SKYFOX、SEVEN CITIES OF GOLD、THE MURDER ON THE ZINDERNEUF、FINANCIAL COOKBOOK、PINBALL CONSTRUCTION SET、MUSIC CONSTRUCTION SET 等等。

由於以上產品之保護方法各有差異，所以爲了更加明瞭解拆的方法，我便將解拆方法一一列出：

AXIS ASSASSIN ——在解拆這程式時，我們必須找出它的保護程式所在，所以我們便要一步步的TRACE ：首先開了開關掣，放入磁碟，隔一會兒在螢幕處會出現LOGO及選擇LEVEL ，選擇完畢後，磁碟機再次讀碟，畫面隨即清除成黑色一片，如果你隻碟在TRACK ＄21及＄22抄得不好，「ERR」 這字樣便會出現在螢幕上，主要原因是PRO—TECTION 是放在TRACK ＄21及＄22處。

爲了找出呼叫保護程式之PROGRAM 是在那一個ADD—RESS處，首先我們必須製造一個「工具」。而製造這工具的方法非常之簡單：第一步是BOOT起一隻用DOS 3.3 FORMAT 而沒有保護的磁碟，繼而是FORMAT 一隻空碟——鍵入INIT H便可。在FORMAT 的中途按RESET 中止磁碟機的運行，這樣你手中的「半FORMAT」之磁碟便成爲我們解拆磁碟的最佳工具，當然有一些PROTECTION 是例外。萬事俱備後，我們便BOOT 起一隻只COPY TRA—CK ＄00—TRACK ＄20之GAME碟，在出現ERROR 及磁碟機停止運行後，放入「工具」磁碟，按RESET （千萬不要POWER OFF)，隨即便會發覺電腦發出「嗶」一聲，及螢幕顯示進入了MONITOR 狀態，這是由於「工具」磁碟沒有DOS 之原因。按C0E8 〔RETURN〕弄停磁碟機，然後鍵入＄AD00L 便可看到PROTECTION 之呼叫程式了。而MAIN PROGRAM 呼叫保護程式是在＄9DE4 處。你只要按＄9DE4L 便發現「JSR＄AD00」 這字樣，解除保護的方法是將＄9DE4 內之＄E6 9D 更改爲＄EF 9D—＄E6 9D 是指＄9DE6的意思，亦即是 JSR＄AD 00之進入點，而＄EF 9D則是剛剛脫離呼叫保護程式的位置，所以便沒有保護存在。找出保護後，便可正正式式製造一個COPY 版本之ASIS ASSASSIN了。首先FORMAT 好一隻磁碟，然後用ULTRA FAST COPY 或 DISK MUNCHER COPY ＄00—20。然後利用NIBBLES AWAY II 之SEC—TOR MODIFY之功能，更改TRACK ＄0D，SECTOR ＄08，在位置＄E4之＄E6 9D 改爲＄EF 9D。在這處只成功了一半，另一半是在TRACK ＄00，SECTOR ＄03 ，在位置＄75之＄D4 62 改爲＄66 60。因爲這隻遊戲之呼叫保護程式共有二次，而＄D4 62是呼叫保護程式的DATA ，而＄66 60則是遊戲進入點的DATA，大家會發覺這些DA—TA 有一些問題，主要原因是這些DATA 是做過手脚的（經過EOR 這功能），如果大家想清楚EOR這功能的話，可以參考一些關於6502 之書籍。當更改完以上各項後，你手上的便眞眞正正成爲COPYA 版本了，不妨一試。

THE LAST GLADIATOR 一鬥獸場這個遊戲曾一度受到大家的歡迎，而這遊戲之保護方法便是日後相繼推出的產品之雛形。同樣找出這遊戲的保護方法是：當POWER ON 後隨即按RESET，CALL—151 進入MONITOR 。鍵入9600<C600. C6FFM〔RETURN〕，然後96F8：A9 80 8D 1C 08 A9 70 8D 25 08 4C 01 08〔RETU—RN〕，以上的改動是將本來原程式應該要將從磁碟讀取之資料存入＄B000—BFFF及＄A000—AFFF改爲＄8000—8FFF及＄7000—7FFF。當＄9600G 運行後，螢幕便會印出＄B003 ，並且磁碟機之紅燈還亮着，按＄C0E8 〔RETURN〕弄停磁碟機，然後鍵入＄A000< 7000. 8FFFM（將程式搬回原本的位置）按＄A800L，便可發現程式的起始位置（STARTING)。而這程式的保護程式存放在＄A00(，而進入保護程式的DATA 是存放在＄A908之03 79 。

E. A. 遊戲軟件保護方法

大家可能不明白7903爲何等如A000呢，原來又是EOR這功能作怪。爲了查明眞相，我們可以輸入以下程式：300：A9 00 49 03 8D D0 03 A9 A0 49 D9 8D D1 03 60，在以上程式中你可發現00 A0這兩個數值，亦即是代表A000的意思。當300G後，在3-D0及3D1處便可發現$03 79 這兩個數值，而在程式中的$AA00處你可以發現「EOR#$03及EOR#$D9」的存在，亦即是說03 79是眞正的代表$A000的意思。保護程式之$A000所保護的方法是COUNT BYTE ，如果結果不相等的話，便會設置進位旗號（SET CARRY FLAT），反之如果相等的話便取消進位旗號（CLEAR CARRY FLAT），然後再由子程式檢定是否有進位旗號設立，有的話便印上ERROR ，相反便繼續運行工作。所以消滅保護的程式，便是將03 79 這DATA改爲67 79，如果你想知道67 79代表什麽位置的話，只要更改以上$300的程式之A9 00爲A9 67及A9 A0 爲79便可，$300G後，便可在$3D0及3D1處找出答案。而這段位置是放在磁碟的TRACK $01，SECTOR $06 ，位置在08處，將03 79改爲67 79。但這還不算是成功的，因爲這程式是有兩個保護程式，而第二個程式是在$A300 處，大家會發覺此段位置與$A000 差不多，其實這兩段程式是一樣的。要消除這段保護只要輸入$A300：A9 00 18 60；更改磁碟在TRACK $01，SECTOR $0C，位置$00。18是消除進位旗號之功能碼（OPCODE)，而整體的含義是消除進位旗號及設置疊積器爲$00來符合PROTECTION之「需求」。大家可能會有一些疑問關於$A300及$A000，既然兩者是一樣的話，爲何$A300可以直接更改爲設置進位置旗號爲00，而$A000不可呢？主要原因是在$A000這段保護程式，當一開始運行程式時（$A800—$AD00）便隨即測試$A000 內之CHECKSUM是否相符，如果不相符便代表$A000內之程式已被人修改，於是便在畫面的左上角印出「？」號並停止程式運行，所以要更改$A000必須要克服CHECKSUM這程式，但這樣做比起更改03 79爲67 79 更爲複雜。而$A300這段程式因沒有受到CHECKSUM牽連，所以便可以直接修改了。

ONE ON ONE——這遊戲的保護與以前一樣，我們只是修改$A300之內容便可。即在TRACK $01，SECTOR $0C，位置$00爲18 A9 00 60，及TRACK $01，SEC-TOR $06，位置$05內之03 79改爲03 65，這處是將原來進入保護程式的$A000之DATA更改爲進入$A300。而同樣地ONE ON ONE亦有兩個保護程式，而另一個是在$BD00處，亦即是在TRACK $0C，SECTOR $04，位置05更改爲18 60便可。

ARCHON ——這是一個鬥智鬥力的遊戲。它的保護方法前半段與ONE ON ONE大致相同，大家只要更改TRA-CK $01，SECTOR $06，位置08之03 79爲61 79。更改完這段後，我們可以玩一會兒，但當遊戲途中再讀碟時便告ERROR 了。所以我們必須將第二部份之保護消除。補充一提是ARCHON這遊戲磁碟之DATA MARK，在TRA-CK $00—TRACK $02是NORMAL的D5 AA AD ，但在TRACK $03—TRACK $22是D5 BB CF ，所以使它成爲COPYA版本，必須將D5 BB CF改爲D5 AA AD。方法是BLOAD ULTRA FAST COPY，然後CALL —151進入MONITOR，輸入80E：D5 BB CF，按803G 運行此程式，當COPY進行時在TRACK $00—TRACK $02及TRACK $05—TRACK $06是會發生ERROR，按「N」便可。COPY完畢後，更改TRACK $02，SECTOR $03，位置47之BB爲AA，位置51之CF 爲AD便可。而第二部份之保護程式，我曾一度找不出來，後來才發現這遊戲是做過手脚的，它將原來的程式用EOR這功能作互拆OR了，然後才WRITE入磁碟內，當讀入回MEMORY 時才再將它還原。找出還原程式只要更改TRACK $12，SECTOR $0C，位置72內之20 08 A0 改爲4C 59 FF。當BOOT 起這隻碟後，選擇完OPTION，一會兒便聽到「嗶」一聲，按$A000L，便看到$00 01 15 62 EA EA EA EA等DATA，而這些DATA是用作EOR之用，將$A064之4C 59 FF還原爲20 08 A0 ，及$A07E改爲4C 59 FF，當$A05AG後便可發現本來如垃圾一般之$6A3C ，一變成爲有條有理的程式，原來這段位置便是第二段保護的所在地，並且可發現它是放在$D800 處的，消除保護只需要輸入$6A3C：2C E8 C0 A2 C8 88 D0 FD CA D0 FA 2C 83 C0 60 ，但由於這些DATA是經過EOR 才儲存到磁碟內的，所以我們將$A000之內容回復原狀—$A000：00 01 15 62 EA EA EA EA ，然後再$A05AG，再$6A3CL，便會看到$6A3C：BA 7E 84 E6 93 D3 26 1B 15 0F 41 97 DF 9C FF 9F，而這些DATA便是經過EOR這功能之後而得來的。修改TRACK $1B，SECTOR $0D，位置3C便告大功告成了。

SKY FOX ——是一個立體遊戲，它的保護和以前有些少不同，以前的保護是檢定進位旗號設置與否來測定磁碟有沒有被人BACKUP，但SKY FOX的保護是測試疊積器內的數值，假若所測試的結果是4D 的話，便是正常，相反便印出ERROR 。我們便可根據此點，製造一隻和原裝碟不遑多讓的複製品了。製造方法：首先如ARCHON一樣，利用了ULTRA FAST COPY來COPY TRACK $00—02 （不要改DATA MARK）然後再COPYTRACK $03—22 （如ARCHON一樣，更改DATA MARK）。當COPY完畢後，更改TRACK $01，SECTOR $06，位置27的03 79爲03 65及TRACK $01，SECTOR $0C，位置00改爲：20 00 A0 A9 4D 18 60 便告完成一部份。再如ARCHON 一樣更改TRACK 02，SECTOR $03之D5 BB CF改爲D5 AA AD。另一部份是在TRACK $0A，SECTOR—$05，位置EB，同樣是測試4D這數值，而保護程式是放在$D700，如果測試結果不等如4D的話便跳到$BFAA 處，而$BFAA的工作是消除記憶及RESET 。爲了不令這部份進行，只需要修改位置EB內容爲：EA EA EA EA EA EA（

E. A. 遊戲軟件保護方法

共6個EA）便可，而EA這OPCODE是NOP （NO OPER—ATION）的意思。但這並未完全成功，還需要再更改一下：TRACK $0A，SECTOR $02，位置FB，將20改爲60（RTS的意思）；TRACK $13，SECTOR $0E，位置3C，更改爲60；TRACK $20，SECTOR $02，位置FB，更改爲60。而以上更改的，全部都是測試ERROR的子程式，如果不符的話，便跳到$BFAA處，所以只要更改爲60（RETURN）來取消這測試的運行。

SEVEN CITIES OF GOLD ——同以上一樣，只要更改TRACK $01，SECTOR $06，位置08之03 79爲62 79。並且更改TRACK $0E，SECTOR $07，位置75之20 00 61爲18 A9 00便算完成解除保護。附帶一提的是遊戲之標題字是在TRACK $0E，SECTOR $0F，你們可以根據此點，修改自己的名字在內。

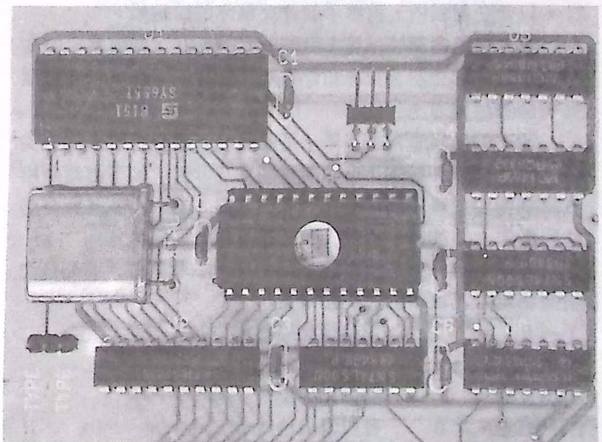
MURDER ON THE ZINDERNEUF ——與SEVEN CITIES OF GOLD大同小異，修改TRACK $01，SECT—OR $06，位置08之03 79爲62 79；及TRACK $01，SECTOR $0C，位置00之4C 68 05改爲18 A9 00 60便可。

FINANCIAL COOKBOOK——這程式之保護非常簡單，只要修改TRACK $01，SECTOR $06，位置08之03 79爲62 79便可以了。

PINBALL CONSTRUCTION SET——這處只適用於「E.A.」出品之NEW VERSION 版本。更改TRACK $03，SECTOR $06，位置8F爲EA EA 便可擁有一個與原裝差不多一模一樣的BACKUP了。

總結來說，「E.A.」之保護方法可說是獨創一格，但如果讀者看過本文後掌握到其中的奧妙，以後對「E.A.」或同類型的保護，便會有一定的破解方法了。■

專業化電腦硬件維修/改良服務



APPLE II或相容機
—主機（HK$80）
—磁碟機（HK$30）
—一般介面咭（HK$25）
—IBM或相容機（HK$250）

＊上述費用不包括零件費
＊詳細收費服務表備索

本會擁有一群豐富專業知識的電腦工程人員，抱著非牟利的宗旨爲廣大的電腦用家提供廉價而專業化的電腦維修保養／改良及其他服務。

本會現招收基本會員，歡迎參加，章程備索。

電子科技學會 ELECTRONIC TECHNOLOGY ASSOCIATION

專改卡仔

收件站：旺角上海街438號同珍商業中心十一字樓四室（電梯十一字）

方尖塔世界爭奪戰

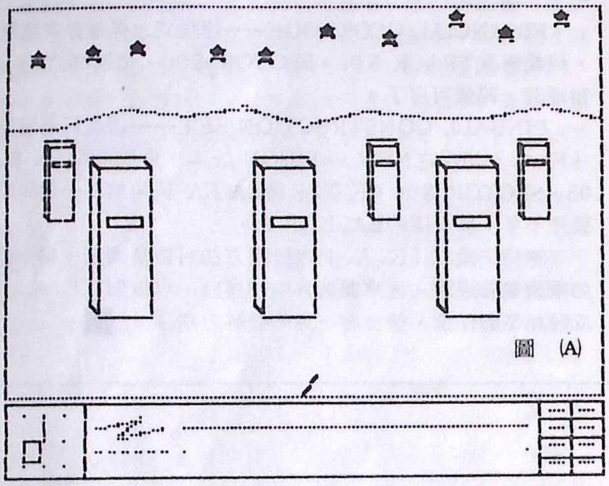


圖 (A)

在一個方尖塔的世界裏，你是一艘探險船的指揮官。當你從宇宙另一個角落執行完探索任務後回來的時候，你發覺你的太空站母艦經被敵人攻佔及控制，敵人駕著母艦四出躲藏，並且在方尖塔裏埋有狙擊手。這些方尖塔裝有雷達感應器，只要在射程內，狙擊手就會攻擊你的探險船。

敵人經常會駕駛一隻小船的偵察飛船在上空盤旋，假如讓它飛越你的上空，它會知會佔領了母艦的同伴將母艦移離另一個隱秘地方。

閣下的任務有兩個，第一是盡量殲滅藏有狙擊手的方尖塔。第二個任務是閣下必須要奪回太空母艦，閣下必須將船置定到雷達螢幕中間的位置（見圖A左下角），並將探測船駛回母艦。假如任務成功，閣下始有機會返回地球。

本遊戲是一個需要動腦筋的動態射擊戰遊戲，下列是控制探測船及其攻擊砲火的指令鍵：

1－8	選擇探測船移動方向
SPACE BAR	探測船向前飛
PADDLE＃0	瞄準雷射槍
BUTTON＃0	發射雷射槍
M　鍵	放射殲滅彈
Q　鍵	終止遊戲
ESC 鍵	檢視任務進行情狀

閣下的探測船擁有兩種攻擊武器：(1)雷射槍及殲滅彈。雷射槍是由PADDLE 0 控制，必須直接瞄準方尖塔前面的控制窗方可摧毀方尖塔內的敵人。殲滅彈則是用M鍵來控制投下。當投射後，在螢幕上所能見到的方尖塔將會全數被殲。不過，探測船只有８枚殲滅彈。圖A中右下角是顯示剩餘殲滅彈的數目。

閣下在這次拯救行動中若能完成任務，將會獲得上頭的嘉許，並獲得晉升。晉升的級數決定於敵人被殲滅的數目（OBELISKS HIT百分比），耗用能源量（ENERGY LEFT百分比），探測船受損程度（DAMAGE LEVEL 百分比）。假如閣下能成功奪回太空母船，並駛返地球，則閣下可額外獲得獎分。

這個遊戲較特出之處是閣下晉升的機會是會視乎在整個戰役中的表現。舉例說，假若閣下在開始時表現得十分勇猛，但到後來却越來越大意，則閣下的得分會由最初一直上升至某個水平，然後下跌。總括來說，要取得最高積分，必定要以消耗最少的能源、殲滅最多的敵人，並且儘量避免探測船受損，與及最終能降落回母船。（有關戰事進行情況可看圖B）

爲了讓大家取得好成績，筆者有幾點要大家特別注意的：

(1) 最近的方尖塔是最具危險性，應該先殲滅它。
(2) 所有移動都會消耗能源，小心計劃好行動。
(3) 要成功降落回母艦，你必須操縱探測船駛到母艦處於螢幕雷達中心的位置，然後駛向前。
(4) 盡可能勿讓敵方偵察飛船駛過上空，否則它會向母艦上的同伴報告你所處的位置，然後將母艦駛離到另一個神秘的地方。
(5) 要降落探測船，小心留意螢幕左下角的雷達方位指示器。只可在母船所處於的方位四份一圓周內移動。

本遊戲只有一個列表，大家可打入後用下列指令存入磁碟： SAVE OBELISK。■

圖 (B)

```
#OBELISKS LEFT= 193    %OBELISKS HIT= 0
# SHOTS FIRED = 0      % DAMAGE      = 2
# MEGABLASTS  = 8      TOTAL RATING = 0
RANK:JR CABIN BOY      HIGHEST TODAY= 0

          *** COMMAND SUMMARY ***

     NUMBERS 1-8 - SET DIRECTION
       SPACE BAR - MOVE FORWARD

      PADDLE # 0 - AIM PHASER
      BUTTON # 0 - FIRE PHASER

        LETTER M - MEGABLAST
        LETTER Q - QUIT MISSION
     <ESC> KEY - MISSION STATUS

        HIT ANY KEY TO CONTINUE
```

方尖塔世界爭奪戰

```
10  REM  ***********************
20  REM  *                     *
30  REM  *     OBELISK         *
40  REM  *                     *
50  REM  * COPYRIGHT (C) 1985  *
60  REM  *                     *
70  REM  * BY  COMPUHOUSE      *
80  REM  *                     *
90  REM  ***********************
100  REM
110  LOMEM: 24576: FOR X = 1 TO 1
   6: READ P,D: NEXT : FOR X =
   800 TO 813: READ Y: POKE X,Y
   : NEXT : FOR X = 832 TO 848:
    READ Y: POKE X,Y: NEXT : POKE
   850,0: POKE 851,0: ONERR  GOTO
   2970
120  GOTO 2990
130  IF DS = [illegible] THEN SC = 3
140  IF DS = 3 THEN SC = 2
150  ON SH GOSUB 160,190,160,160,
   200,210,220,230,190,240,190:
    RETURN
160 Z = A + SC * (SC + 1):W = B +
   8 - SC
170  HPLOT Z,W TO Z + 12 * SC,W TO
   Z + 12 * SC,W + 30 * SC TO Z
   ,W + 30 * SC TO Z,W TO Z - 2
    * SC,W - 2 * SC TO Z + 10 *
   SC,W - 2 * SC TO Z + 12 * SC
   ,W: HPLOT Z - 2 * SC,W - 2 *
   SC TO Z - 2 * SC,W + 28 * SC
    TO Z,W + 30 * SC
180  HPLOT Z + 2 * SC,B + 13 * SC
    TO Z + 10 * SC,B + 13 * SC TO
   Z + 10 * SC,B + 15 * SC TO Z
    + 2 * SC,B + 15 * SC TO Z +
   2 * SC,B + 13 * SC: RETURN
190  HPLOT A - 5 * SC,B + 20 * SC
    TO A,B TO A + 5 * SC,B + 20
    * SC TO A,B + 15 * SC TO A -
   5 * SC,B + 20 * SC: HPLOT A,
   B + 12 * SC TO A,B + 20 * SC
   : RETURN
200  HPLOT A - 3,B + 2 TO A + 3,B
    + 2 TO A - 3,B + 6 TO A,B TO
   A + 3,B + 6 TO A - 3,B + 2: RETURN
210  HPLOT A,B + 7 TO A + 20,B +
   7 TO A + 10,B TO A,B + 7: RETURN
220 W = B - 5:ZZ = A - 5: HPLOT Z
   Z,W TO ZZ + 8,W + 6 TO ZZ +
   5,W TO ZZ + 9,W + 5 TO ZZ,W +
   9 TO ZZ + 8,W: RETURN
230  HPLOT A,B + 2 TO A + 14 * SC
   ,B + 2: HPLOT A + 3 * SC,B +
   1 TO A + 11 * SC,B + 1: HPLOT
   A + 5 * SC,B TO A + 9 * SC,B
   : RETURN
240  HPLOT A + 1,B + 1 TO A + 9,B
    + 1 TO A + 9,B + 9 TO A + 1
   ,B + 9 TO A + 1,B + 1: RETURN
250 OP = PD:PD =  PDL (0): IF OP <
    > PD THEN  HCOLOR= 0: HPLOT
   140,158 TO LX,150:LX = 127 +
    INT (PD / 255 * 25): HCOLOR=
   3: HPLOT 140,158 TO LX,150:S
   I = LX - 140
260  RETURN
270 X =  INT ( RND (1) * 10): IF
   RP(X) < > 1 THEN 400
280 DD = 400: ON X GOTO 290,300,3
   10,320,330,340,350,360,370
290 XR = 135:YR = 62:Z = 135:DD =
   800: GOTO 380
300 XR = 135:YR = 70:Z = 135:DD =
   600: GOTO 380
310 XR = 135:YR = 55:Z = 135: GOTO
   380
320 XR = 60:YR = 70:Z = 64: GOTO
   380
330 XR = 220:YR = 70:Z = 216: GOTO
   380
340 XR = 28:YR = 55:Z = 32: GOTO
   380
350 XR = 93:YR = 55:Z = 97: GOTO
   380
360 XR = 178:YR = 55:Z = 174: GOTO
   380
370 XR = 248:YR = 55:Z = 244
380  IF  PEEK ( - 16287) > 127 THEN
   420
390  POKE 806,25: POKE 801,20: HCOLOR=
   5: CALL 800: HPLOT XR,YR TO
   Z,158:SH = 7:A = Z:B = 153: GOSUB
   220: HCOLOR= 0: POKE 839,10:
    POKE 845,40: CALL 832: HPLOT
   XR,YR TO Z,158: GOSUB 220: HCOLOR=
   3:DA = DA + DD: GOSUB 920
400  IF DA > 40000 THEN EN = EN -
   50
410  GOSUB 250:A$ = "":FR =  FRE
   (0): IF  PEEK ( - 16287) < 1
   28 THEN 470
420  HCOLOR= 6: POKE 839,0: POKE
   845,60: CALL 832:FX = LX:FY =
   150:SF = SF + 1:OX = FX:OY =
   FY:FY = P3:DS = 1:Y = FY:FX =
   LX + SI * (150 - FY) / P5: IF
   FX > 1 AND FX < 267 THEN  HPLOT
   OX,OY TO FX,FY:OX = FX:OY =
   FY: GOTO 1940
430  GOTO 460
440 FY = P2:DS = 3: GOTO 1970
450 FY = P1:DS = 4:Y = FY:FX = LX
    + SI * (150 - FY) / P5: IF
   FX > 1 AND FX < 267 THEN  HPLOT
   OX,OY TO FX,FY:OX = FX:OY =
   FY: GOTO 2020
460 EN = EN - 500: GOSUB 900: HCOLOR=
   0: HPLOT LX,150 TO OX,OY: GOSUB
   250
470  IF  PEEK ( - 16287) > 127 THEN
   420
```

```
480  IF  RND (1) > .96 THEN 700
490 ADJ = 1:KEY =  PEEK (49152): IF
    KEY < 128 THEN 270
500  GET A$: IF A$ = " " THEN 143
    0
510  IF A$ <  > "M" OR ME <  = 0 THEN
    640
520 ME = ME - 1:EN = EN - 1600: POKE
    839,0: POKE 845,50: FOR Z =
    1 TO 2: CALL 832: HPLOT 0,0:
     CALL 62454: HCOLOR= 2: CALL
    832: HPLOT 0,0: CALL 62454:
    HCOLOR= 5: NEXT : CALL 832
530 F2 = 1: FOR Z = 1 TO 9: ON Z GOTO
    540,550,560,570,580,590,600,
    610,620
540 FX = 131:FY = P3: GOTO 1940
550 FX = 131:FY = P2: GOTO 1970
560 FX = 128:FY = P1: GOTO 2020
570 FX = 59:FY = P2: GOTO 1970
580 FX = 221: GOTO 1970
590 FX = 32:FY = P1: GOTO 2020
600 FX = 92: GOTO 2020
610 FX = 176: GOTO 2020
620 FX = 248: GOTO 2020
630  NEXT :F2 =  PEEK (49168):F2 =
    0: GOTO 1160
640  IF A$ =  CHR$ (27) THEN T1$ =
    " STATUS REPORT ": POKE 801,
    0: POKE 806,8: CALL 800: GOSUB
    2670: CALL 800
650  IF  VAL (A$) > 0 AND  VAL (A
    $) < 9 THEN  HCOLOR= 0: GOSUB
    1260:DI =  VAL (A$): GOSUB 1
    210
660  IF A$ <  > "Q" THEN 270
670  TEXT : HOME : VTAB (12): HTAB
    (4): FLASH : PRINT "DO YOU R
    EALLY WANT TO QUIT <Y/N>?";:
     FOR X = 1 TO 3: POKE 801,0:
     POKE 806,10: CALL 800: POKE
    801,200: POKE 806,15: CALL 8
    00: NEXT
680  NORMAL : GET AN$: IF AN$ <  >
    "Y" THEN  POKE  - 16302,0: POKE
     - 16299,0: POKE  - 16304,0:
     GOTO 270
690 EOG = 1:T1$ = "*QUIT MISSION*
    ":ADJ = .8: GOTO 2670
700 X = 241:I =  - 8 -  INT ( RND
    (1) * 10):R =  RND (1):FF =
    1: IF R > .5 THEN I = I * -
    1:X = 20
710 Y = 28: HCOLOR= 3: POKE 801,6
    0: POKE 806,2
720  HCOLOR= 1:X = X + I:A = X:B =
    Y: GOSUB 210: CALL 800
730  IF  RND (1) < .30 THEN  HCOLOR=
    3: HPLOT X + P4,Y + P4 TO X +
    P4,158: HCOLOR= 5:A = X + 9:
    B = 152: GOSUB 220: CALL 800
    : HCOLOR= 0: HPLOT X + P4,Y +
    P4 TO X + P4,158: GOSUB 220:
     HCOLOR= 3:DA = DA + 600
740  CALL 800: HCOLOR= 0:A = X:B =
    Y: GOSUB 210: IF X > 19 AND
    X < 242 THEN 770
750 X =  PEEK (49168): FOR X = 1 TO
    2: POKE 801,0: POKE 806,10: CALL
    800: POKE 801,200: POKE 806,
    15: CALL 800: NEXT :TR = TR +
    1
760 X = R1:Y = R2:F2 = 1:C$ = "1"
    : GOSUB 2570:F2 = 0:R1 =  INT
    ( RND (1) * 16) + 2:R2 =  INT
    ( RND (1) * 16) + 2:P$(R2) =
    LEFT$ (P$(R2),R1 - 1) + "3"
    + RIGHT$ (P$(R2),20 - R1):
    MV = 0:PD = 0:F1 = 0:S1 = 0:
     GOTO 1160
770  IF FF =  - 1 THEN  HCOLOR= P
    6: GOTO 830
780 OP = PD:PD =  PDL (0): IF OP <
     > PD THEN  HCOLOR= 0: HPLOT
    140,158 TO LX,150:LX = 127 +
     INT (PD / 255 * 25): HCOLOR=
    3: HPLOT 140,158 TO LX,150:S
    I = LX - 140
790  IF  PEEK ( - 16287) < 128 THEN
    720
800 EN = EN - 450:F1 = 1: HCOLOR=
    6: POKE 839,0: POKE 845,60: CALL
    832:OX = LX:OY = 150:SF = SF
     + 1: IF LX < 131 OR LX > 14
    9 THEN FY = 78: GOTO 820
810 FY = 62
820 FX = LX + SI * (150 - FY) / P
    5: IF FF = 1 THEN 860
830  IF LX < 132 OR LX > 147 THEN
    FF = 1: GOTO 870
840 OY = FY:OX = FX:FY = 30:FX =
    LX + SI * (150 - FY) / P5: IF
    FX < 2 THEN FX = 2
850  IF FX > 266 THEN FX = 266
860  HPLOT OX,OY TO FX,FY:FF = FF
     *  - 1: IF FF =  - 1 THEN 7
    20
870  HCOLOR= 0:LY = 150: HPLOT LX
    ,LY TO FX,FY: IF FY = 30 AND
    FX < X + 15 AND FX > X + 3 THEN
    890
880 F1 = 0: GOTO 720
890 DS = 4:S = 1: GOSUB 920: GOSUB
    900: GOTO 2100
900  IF EN > 0 THEN  HCOLOR= 0: HPLOT
    242,171 TO 85 + EN / 99999 *
    157,171: HCOLOR= 6: RETURN
910 T1$ = " OUT OF ENERGY ":EOG =
    1:EN = 0: GOTO 2670
920  IF DA < 99999 THEN  HCOLOR=
    5: HPLOT 85,181 TO 85 + DA /
    99999 * 158,181: HCOLOR= 3: RETURN
930 T1$ = "CRITICAL DAMAGE":EOG =
    1:DA = 99999: GOTO 2670
```

```
940 IF X < 1 THEN X = 20 + X
950 IF X > 20 THEN X = X - 20
960 IF Y < 1 THEN Y = 20 + Y
970 IF Y > 20 THEN Y = Y - 20
980 LC = VAL ( MID$ (P$(Y),X,1))
    : RETURN
990 IF LC = 1 THEN RP(P) = 1:SC =
    1: HCOLOR= 3:SH = 1:A = XL:B
    = YL: GOSUB 130: RETURN
1000 IF LC = 2 AND DS = 1 THEN S
    C = 3:RX = 111:RY = 148
1010 IF LC = 2 AND DS = 4 THEN S
    C = 1:RX = XL - 4:RY = YL +
    25
1020 IF LC = 2 AND DS = 3 THEN S
    C = 2:RX = XL + 8:RY = YL +
    70
1030 IF LC = 2 THEN HCOLOR= 5:S
    H = 8:A = RX:B = RY: GOSUB 1
    30
1040 IF LC = 3 AND DS = 4 THEN S
    H = 9:A = XL:B = YL:SC = 1: GOSUB
    130
1050 IF LC = 3 AND DS = 3 THEN S
    H = 2:SC = 2:A = XL + 15:B =
    YL: GOSUB 130
1060 IF LC = 3 AND DS = 1 THEN S
    C = 3:SH = 2:A = 140:B = 95:
    GOSUB 130: HCOLOR= 0: HPLOT
    50,160 TO 225,160
1070 HCOLOR= 3:SC = 1: RETURN
1080 CLEAR : ONERR GOTO 2970
1090 DIM P$(20):DI = 1:A$ = "":
    LX = 140:EN = 99999:ME = 8:P
    1 = 62:P2 = 78:P3 = 86:P4 =
    10:P5 = 8:P6 = 6:P7 = 33:ADJ
    = 1
1100 VV = ( PEEK (78) + 1) / 256:
    FOR Y = 1 TO 20: FOR X = 1 TO
    20:V = RND (1) * VV: IF V <
    .50 * VV THEN V = 1:TR = TR +
    1
1110 V$ = STR$ ( INT (V)):P$(X) =
    P$(X) + V$: NEXT X: POKE 801
    ,10 + Y: POKE 806,25 - Y: CALL
    800: NEXT Y:R = RND (1) * 4
    :RL = TR:V = INT (V * 20):V
    V = INT (VV * 20): IF VV <
    2 THEN VV = 2
1120 IF VV > 19 THEN VV = 19
1130 P$(V) = LEFT$ (P$(V),VV - 1
    ) + "3"+ RIGHT$ (P$(V),20 -
    VV):R1 = VV:R2 = V:PX = INT
    (VV + 8 + R):PY = INT (V +
    8 + R): IF PX > 20 THEN PX =
    PX - 20
1140 IF PY > 20 THEN PY = PY - 2
    0
1150 OL = 1: GOTO 1430
1160 HGR2 : POKE - 16302,0:PD =
    - 1: HCOLOR= 1: HPLOT 1,0 TO
    279,0 TO 279,159 TO 1,159 TO
    1,0
1170 HCOLOR= 2:HX = 1: HPLOT HX,
    50:HX = 0: FOR X = 1 TO 10:H
    Y = INT ( RND (1) * 10):HX =
    HX + 28: HPLOT TO HX - 1,50
    - HY: NEXT : HCOLOR= 1: HPLOT
    HX - 1,50 - HY: HCOLOR= 3
1180 FR = FRE (0): HCOLOR= 3: HPLOT
    1,50: FOR X = 28 TO 280 STEP
    28:B = 22 - INT ( RND (1) *
    22):SH = 5:A = X - 20 + ( INT
    ( RND (1) * 10)): GOSUB 200:
    NEXT : HCOLOR= 3
1190 HCOLOR= 3: HPLOT 1,161 TO 2
    79,161 TO 279,191 TO 1,191 TO
    1,161: HPLOT 40,161 TO 40,19
    1: HPLOT 249,161 TO 249,191:
    HPLOT 264,161 TO 264,191: HPLOT
    250,168 TO 278,168: HPLOT 25
    0,176 TO 278,176: HPLOT 250,
    184 TO 278,184
1200 HCOLOR= 3: HPLOT 20,166: HPLOT
    30,166: HPLOT 30,176: HPLOT
    30,186: HPLOT 20,186: HPLOT
    10,186: HPLOT 10,176: HPLOT
    10,166: GOTO 1350
1210 HCOLOR= 0:SH = 10:A = 9:B =
    165: GOSUB 240:B = 175: GOSUB
    240:A = 18:B = 165: GOSUB 24
    0:B = 175: GOSUB 240: HCOLOR=
    3: HPLOT 20,166: HPLOT 30,16
    6: HPLOT 30,176: HPLOT 30,18
    6: HPLOT 20,186: HPLOT 10,18
    6: HPLOT 10,176: HPLOT 10,16
    6
1220 X = PX - R1:Y = PY - R2: IF
    X = > 0 AND Y = > 0 THEN S
    H = 10:A = 9:B = 165: GOSUB
    240: GOTO 1260
1230 IF X = > 0 AND Y < 0 THEN
    SH = 10:A = 9:B = 175: GOSUB
    240: GOTO 1260
1240 IF X < 0 AND Y = > 0 THEN
    SH = 10:A = 18:B = 165: GOSUB
    240: GOTO 1260
1250 SH = 10:A = 18:B = 175: GOSUB
    240
1260 ON DI GOSUB 1270,1280,1290,
    1300,1310,1320,1330,1340: RETURN
1270 HPLOT 20,176 TO 20,166: RETURN
1280 HPLOT 20,176 TO 30,166: RETURN
1290 HPLOT 20,176 TO 30,176: RETURN
1300 HPLOT 20,176 TO 30,186: RETURN
1310 HPLOT 20,176 TO 20,186: RETURN
1320 HPLOT 20,176 TO 10,186: RETURN
```

```
1330  HPLOT 20,176 TO 10,176: RETURN
1340  HPLOT 20,176 TO 10,166: RETURN
1350 Y = 171: HCOLOR= 1: HPLOT 45
     ,Y TO 65,167 TO 55,175 TO 75
     ,Y:X = 84 + EN / 99999 * 159
     : HPLOT 85,Y TO X,Y: HPLOT 2
     43,171
1360 Y = 181: HCOLOR= 5: FOR X =
     45 TO 75 STEP 6: HPLOT X,Y TO
     X + 3,Y: NEXT :X = 85 + DA /
     99999 * 158: HPLOT 85,Y TO X
     ,Y: HPLOT 243,Y: IF ME < 1 THEN
     1390
1370  HCOLOR= 6:X1 = 253:X2 = 260
     :Y = 165:E = ME: IF ME > 4 THEN
     E = 4
1380  FOR X = 1 TO E: HPLOT X1,Y TO
     X2,Y:Y = Y + 7: NEXT
1390  IF ME < 5 THEN  HCOLOR= 3: GOTO
     1410
1400 X1 = 268:X2 = 275:Y = 165:E =
     ME - 4: FOR X = 1 TO E: HPLOT
     X1,Y TO X2,Y:Y = Y + 7: NEXT
     : HCOLOR= 3: IF F5 = 1 THEN
     RETURN
1410  IF S1 = 1 THEN  GOSUB 1520:
     GOSUB 1210: GOTO 270
1420  GOSUB 1530: GOSUB 1210: GOTO
     270
1430  POKE 801,30: POKE 806,25: CALL
     800:EN = EN - 300: ON DI GOTO
     1440,1450,1460,1470,1480,149
     0,1500,1510
1440 XI = 0:YI = 1: GOTO 1160
1450 XI = 1:YI = 1: GOTO 1160
1460 XI = 1:YI = 0: GOTO 1160
1470 XI = 1:YI =  - 1: GOTO 1160
1480 XI = 0:YI =  - 1: GOTO 1160
1490 XI =  - 1:YI =  - 1: GOTO 11
     60
1500 XI =  - 1:YI = 0: GOTO 1160
1510 XI =  - 1:YI = 1: GOTO 1160
1520  IF A$ <  > "M" THEN PX = PX
     + XI:PY = PY - YI
1530  FOR X = 1 TO 9:RP(X) = 0: NEXT
     : IF PX < 1 THEN PX = 20
1540  IF PX > 20 THEN PX = 1
1550  IF PY < 1 THEN PY = 20
1560  IF PY > 20 THEN PY = 1
1570  IF S1 = 0 THEN S1 = 1: GOTO
     1620
1580  IF A$ <  > "M" THEN X = PX:
     Y = PY: GOSUB 980: IF LC = 1
     AND DP = DI THEN DA = DA +
     10000: TEXT : HOME : VTAB 10
     : INVERSE : PRINT "ROVER HIT
     OBELISK--- IT SELF DESTRUCT
     S": NORMAL : PRINT : FLASH :
     PRINT "<< ROVER SUSTAINS SI
     GNIFICANT DAMAGE >>"
1590  IF A$ <  > "M" AND LC = 1 AND
     DP = DI THEN  NORMAL : FOR X
     = 1 TO 3: POKE 801,0: POKE
     806,10: CALL 800: POKE 801,2
     00: POKE 806,15: CALL 800: NEXT
     : FOR X = 1 TO 1500: NEXT : POKE
     - 16302,0: POKE  - 16299,0:
     POKE  - 16304,0
1600  IF LC = 3 AND DP <  > DI THEN
     1620
1610  IF LC = 3 THEN T1$ = " SAFE
     IN SHIP! ":EOG = 1:ADJ = 1.
     2: GOTO 2940
1620 DP = DI:X = PX + 1 * XI:Y =
     PY - 1 * YI: GOSUB 940: IF L
     C <  > 0 THEN XL = 100:YL =
     50:DS = 1:P = 1: GOSUB 990: IF
     LC <  > 2 THEN 1650
1630 X = PX + 2 * XI:Y = PY - 2 *
     YI: GOSUB 940: IF LC <  > 0 THEN
     XL = 115:YL = 60:DS = 3:P =
     2: GOSUB 990: IF LC <  > 2 THEN
     1650
1640 X = PX + 3 * XI:Y = PY - 3 *
     YI: GOSUB 940: IF LC <  > 0 THEN
     XL = 127:YL = 50:DS = 4:P =
     3: GOSUB 990
1650  ON DI GOTO 1660,1730,1800,1
     870,1660,1730,1800,1870
1660 YL = 60:DS = 3:X = PX - 1:Y =
     PY - 2 * YI: GOSUB 940: IF L
     C <  > 0 THEN XL = 40:P = 4:
     GOSUB 990
1670 X = PX + 1: GOSUB 940: IF LC
     <  > 0 THEN XL = 200:P = 5:
     GOSUB 990
1680 YL = 50:DS = 4:X = PX - 2:Y =
     PY  3 * YI: GOSUB 940: IF L
     C <  > 0 THEN XL = 20:P = 6:
     GOSUB 990
1690 X = PX + 1: GOSUB 940: IF LC
     <  > 0 THEN XL = 85:P = 7: GOSUB
     990
1700 X = PX + 1: GOSUB 940: IF LC
     <  > 0 THEN XL = 170:P = 8:
     GOSUB 990
1710 X = PX + 2: GOSUB 940: IF LC
     <  > 0 THEN XL = 240:P = 9:
     GOSUB 990
1720  RETURN
1730 YL = 60:DS = 3:X = PX + 1 *
     XI:Y = PY - 2 * YI: GOSUB 94
     0: IF LC <  > 0 THEN XL = 40
     :P = 4: GOSUB 990
1740 X = PX + 2 * XI:Y = PY - 1 *
     YI: GOSUB 940: IF LC <  > 0 THEN
     XL = 200:P = 5: GOSUB 990
1750 YL = 50:DS = 4:X = PX + 1 *
     XI:Y = PY - 3 * YI: GOSUB 94
     0: IF LC <  > 0 THEN XL = 20
     :P = 6: GOSUB 990
```

```
1760 X = PX + 2 * XI: GOSUB 940: IF LC < > 0 THEN XL = 85:P = 7: GOSUB 990
1770 X = PX + 3 * XI:Y = PY - 2 * YI: GOSUB 940: IF LC < > 0 THEN XL = 170:P = 8: GOSUB 990
1780 Y = PY - 1 * YI: GOSUB 940: IF LC < > 0 THEN XL = 240:P = 9: GOSUB 990
1790 RETURN
1800 YL = 60:DS = 3:X = PX + XI * 2:Y = PY + 1: GOSUB 940: IF LC < > 0 THEN XL = 40:P = 4: GOSUB 990
1810 Y = PY - 1: GOSUB 940: IF LC < > 0 THEN XL = 200:P = 5: GOSUB 990
1820 YL = 50:DS = 4:X = PX + XI * 3:Y = PY + 2: GOSUB 940: IF LC < > 0 THEN XL = 20:P = 6: GOSUB 990
1830 Y = PY + 1: GOSUB 940: IF LC < > 0 THEN XL = 85:P = 7: GOSUB 990
1840 Y = PY - 1: GOSUB 940: IF LC < > 0 THEN XL = 170:P = 8: GOSUB 990
1850 X = PX + XI * 3:Y = PY - 2: GOSUB 940: IF LC < > 0 THEN XL = 240:P = 9: GOSUB 990
1860 RETURN
1870 YL = 60:DS = 3:X = PX + 2 * XI:Y = PY - 1 * YI: GOSUB 940: IF LC < > 0 THEN XL = 40:P = 4: GOSUB 990
1880 X = PX + 1 * XI:Y = PY - 2 * YI: GOSUB 940: IF LC < > 0 THEN XL = 200:P = 5: GOSUB 990
1890 YL = 50:DS = 4:X = PX + 3 * XI:Y = PY - 1 * YI: GOSUB 940: IF LC < > 0 THEN XL = 20:P = 6: GOSUB 990
1900 Y = PY - 2 * YI: GOSUB 940: IF LC < > 0 THEN XL = 85:P = 7: GOSUB 990
1910 X = PX + 2 * XI:Y = PY - 3 * YI: GOSUB 940: IF LC < > 0 THEN XL = 170:P = 8: GOSUB 990
1920 X = PX + 1 * XI:Y = PY - 3 * YI: GOSUB 940: IF LC < > 0 THEN XL = 240:P = 9: GOSUB 990
1930 RETURN
1940 IF FX = 131 AND RP(1) = 1 THEN X = 131:XL = 100:YL = 50:RX = 111:RY = 148:RP(1) = 0: GOTO 2090
1950 IF F2 = 1 THEN 630
1960 GOTO 440
1970 IF FX = 59 AND RP(4) = 1 THEN X = 56:XL = 40:YL = 60:RX = 48:RY = 130:RP(4) = 0: GOTO 2090
1980 IF FX = 131 AND RP(2) = 1 THEN X = 131:XL = 115:YL = 60:RX = 123:RY = 130:RP(2) = 0: GOTO 2090
1990 IF FX = 221 AND RP(5) = 1 THEN X = 216:XL = 200:YL = 60:RX = 208:RY = 130:RP(5) = 0: GOTO 2090
2000 IF FY < > P1 AND F2 = 0 THEN 450
2010 GOTO 630
2020 IF FX = 32 AND RP(6) = 1 THEN X = 24:XL = 20:YL = 50:RX = 16:RY = 75:RP(6) = 0: GOTO 2090
2030 IF FX = 92 AND RP(7) = 1 THEN X = 89:XL = 85:YL = 50:RX = 81:RY = 75:RP(7) = 0: GOTO 2090
2040 IF FX = 128 AND RP(3) = 1 THEN X = 131:XL = 127:YL = 50:RX = 123:RY = 75:RP(3) = 0: GOTO 2090
2050 IF FX = 176 AND RP(8) = 1 THEN X = 174:XL = 170:YL = 50:RX = 166:RY = 75:RP(8) = 0: GOTO 2090
2060 IF FX = 248 AND RP(9) = 1 THEN X = 244:XL = 240:YL = 50:RX = 236:RY = 75:RP(9) = 0: GOTO 2090
2070 IF F2 = 1 THEN 630
2080 GOTO 460
2090 RL = RL   1: IF F2 = 1 THEN GOSUB 2150: GOTO 630
2100 HCOLOR= 0: HPLOT LX,150 TO OX,OY: FOR S = 1 TO 6 - DS:SC = S:XX = X + 5:YY = Y + 5: POKE 839,40: POKE 845,8: HCOLOR= 5: CALL 832:SH = 7:A = XX:B = YY: GOSUB 220: HCOLOR= 0: POKE 839,30: CALL 832: GOSUB 220: NEXT :SC = 1: HCOLOR= 3
2110 IF F1 = 1 THEN F1 = 0: GOTO 270
2120 HCOLOR= 5:SH = DS:A = XL:B = YL: GOSUB 130: HCOLOR= 0: GOSUB 130: IF DS = 1 THEN SC = 3
2130 GOSUB 2150: HCOLOR= 5: IF P > 2 THEN SH = 8:A = RX:B = RY: GOSUB 130: HCOLOR= 3
2140 SC = 1: HCOLOR= 3: GOTO 270
2150 C$ = "2": IF YL = 50 AND XL = 100 THEN GOSUB 2240: GOTO 2570
2160 IF YL = 60 AND XL = 115 THEN GOSUB 2250: GOTO 2570
2170 IF YL = 50 AND XL = 127 THEN GOSUB 2260: GOTO 2570
2180 IF YL = 60 AND XL = 40 THEN GOSUB 2270: GOTO 2570
```

```
2190  IF YL = 60 AND XL = 200 THEN GOSUB 2320: GOTO 2570
2200  IF YL = 50 AND XL = 20 THEN GOSUB 2370: GOTO 2570
2210  IF YL = 50 AND XL = 85 THEN GOSUB 2420: GOTO 2570
2220  IF YL = 50 AND XL = 170 THEN GOSUB 2470: GOTO 2570
2230  IF YL = 50 AND XL = 240 THEN GOSUB 2520: GOTO 2570
2240 X = PX + 1 * XI:Y = PY - 1 * YI: RETURN
2250 X = PX + 2 * XI:Y = PY - 2 * YI: RETURN
2260 X = PX + 3 * XI:Y = PY - 3 * YI: RETURN
2270  ON DI GOTO 2280,2290,2300,2310,2280,2290,2300,2310
2280 X = PX - 1:Y = PY - 2 * YI: RETURN
2290 X = PX + 1 * XI:Y = PY - 2 * YI: RETURN
2300 X = PX + 2 * XI:Y = PY + 1: RETURN
2310 X = PX + 2 * XI:Y = PY - 1 * YI: RETURN
2320  ON DI GOTO 2330,2340,2350,2360,2330,2340,2350,2360
2330 X = PX + 1:Y = PY - 2 * YI: RETURN
2340 X = PX + 2 * XI:Y = PY - 1 * YI: RETURN
2350 X = PX + 2 * XI:Y = PY - 1: RETURN
2360 X = PX + 1 * XI:Y = PY - 2 * YI: RETURN
2370  ON DI GOTO 2380,2390,2400,2410,2380,2390,2400,2410
2380 X = PX - 2:Y = PY - 3 * YI: RETURN
2390 X = PX + 1 * XI:Y = PY - 3 * YI: RETURN
2400 X = PX + 3 * XI:Y = PY + 2: RETURN
2410 X = PX + 3 * XI:Y = PY - 1 * YI: RETURN
2420  ON DI GOTO 2430,2440,2450,2460,2430,2440,2450,2460
2430 X = PX - 1:Y = PY - 3 * YI: RETURN
2440 X = PX + 2 * XI:Y = PY - 3 * YI: RETURN
2450 X = PX + 3 * XI:Y = PY + 1: RETURN
2460 X = PX + 3 * XI:Y = PY - 2 * YI: RETURN
2470  ON DI GOTO 2480,2490,2500,2510,2480,2490,2500,2510
2480 X = PX + 1:Y = PY - 3 * YI: RETURN
2490 X = PX + 3 * XI:Y = PY - 2 * YI: RETURN
2500 X = PX + 3 * XI:Y = PY - 1: RETURN
2510 X = PX + 2 * XI:Y = PY - 3 * YI: RETURN
2520  ON DI GOTO 2530,2540,2550,2560,2530,2540,2550,2560
2530 X = PX + 2:Y = PY - 3 * YI: RETURN
2540 X = PX + 3 * XI:Y = PY - 1 * YI: RETURN
2550 X = PX + 3 * XI:Y = PY - 2: RETURN
2560 X = PX + 1 * XI:Y = PY - 3 * YI: RETURN
2570  GOSUB 940: IF X < 2 THEN P$(Y) = C$ + RIGHT$ (P$(Y),19): RETURN
2580  IF X > 19 THEN P$(Y) = LEFT$ (P$(Y),19) + C$: RETURN
2590 P$(Y) = LEFT$ (P$(Y),X - 1) + C$ + RIGHT$ (P$(Y),20 - X): IF F2 = 1 THEN RETURN
2600  IF FY = P3 THEN 2630
2610  IF FY = P2 AND FX = 131 THEN 2640
2620 P = 3: RETURN
2630 X = PX + 1 * XI:Y = PY - 1 * YI: GOSUB 940: IF LC < > 0 THEN XL = 100:YL = 50:DS = 1:P = 1: GOSUB 990: IF LC < > 2 THEN RETURN
2640 X = PX + 2 * XI:Y = PY - 2 * YI:P = 2: GOSUB 940: IF LC < > 0 THEN XL = 115:YL = 60:DS = 3: GOSUB 990: IF LC < > 2 THEN RETURN
2650 X = PX + 3 * XI:Y = PY - 3 * YI: GOSUB 940: IF LC < > 0 THEN XL = 127:YL = 50:DS = 4:P = 3: GOSUB 990
2660 P = 2: RETURN
2670 EL = INT (EN / 99999 * 100 + .5):RH = INT ((TR - RL + .000001) / (TR + .001) * 100 + .5):DM = INT ((DA + .000001) / 99999 * 100 + .5):HS = PEEK (850) + PEEK (851) * 256
2680 R = INT (ADJ * RH * ((200 + EL) / 100) - DM): IF ADJ = .8 THEN R = R - 20
2690  IF ADJ = 1.2 THEN R = R + 20
2700  IF R < 0 THEN R = 0
2710  IF R < HS OR EOG = 0 THEN F8 = 0: GOTO 2730
2720 HS = R: POKE 851,HS / 256: POKE 850,HS - PEEK (851) * 256:F8 = 1
2730 TEXT : HOME : HTAB (13): FLASH : PRINT T1$: NORMAL : PRINT : IF EOG = 1 THEN POKE 839,0: POKE 845,50: FOR X = 1 TO 5: CALL 832: NEXT
2740 HTAB (9): INVERSE : PRINT "*** MISSION STATUS ***": NORMAL : PRINT : PRINT "CURR DIRECTION= ";DI;" % ENERGY LEFT = ";EL: PRINT "#OBELISKS LEF
```

```
T= ";RL;: PRINT  TAB( 22)"%O
BELISKS HIT= ";RH
2750  PRINT "# SHOTS FIRED = ";SF
;: PRINT  TAB( 22)"% DAMAGE
   = ";DM: PRINT "# MEGABLA
STS  = ";ME;: PRINT  TAB( 22
)"TOTAL RATING = ";R: GOSUB
2850: REM  COMPUTE RANK
2760  PRINT "RANK:";R$;: HTAB (22
): IF F8 = 1 THEN  FLASH
2770  PRINT "HIGHEST TODAY= ";HS:
 NORMAL : PRINT : HTAB (9): INVERSE :
 PRINT "*** COMMAND SUMMARY ***
": NORMAL : PRINT : PRINT  TAB(
7)"NUMBERS 1-8 - SET DIRECTI
ON": PRINT  TAB( 9)"SPACE BA
R - MOVE FORWARD": PRINT
2780  PRINT  TAB( 8)"PADDLE # 0 -
 AIM PHASER": PRINT  TAB( 8)
"BUTTON # 0 - FIRE PHASER": PRINT
: PRINT  TAB( 10)"LETTER M -
 MEGABLAST": PRINT  TAB( 10)
"LETTER Q - QUIT MISSION": PRINT
 TAB( 9)"<ESC> KEY - MISSION
 STATUS"
2790  IF  NOT EOG THEN 2840
2800  GOSUB 2930: VTAB (24): HTAB
 (9): PRINT "PLAY ANOTHER GAM
E <Y/N>?";
2810  GET AN$: IF AN$ <  > "Y" AND
AN$ <  > "N" THEN 2810
2820  IF AN$ = "Y" THEN OL = 0:NX
 = 1: GOTO 2990
2830  VTAB (23): HTAB 1: CALL  -
958: HTAB 9: INVERSE : PRINT
"<< THANKS FOR PLAYING >>";:
 NORMAL : END
2840  VTAB (24): PRINT  TAB( 9)"H
IT ANY KEY TO CONTINUE";: GET
AN$: POKE  - 16302,0: POKE  -
16299,0: POKE  - 16304,0: RETURN

2850  IF R < 16 THEN R$ = "JR CAB
IN BOY": RETURN
2860  IF R < 31 THEN R$ = "ROVER
MECHANIC": RETURN
2870  IF R < 51 THEN R$ = "ROVER
CREWMAN": RETURN
2880  IF R < 76 THEN R$ = "JR STA
R PILOT": RETURN
2890  IF R < 96 THEN R$ = "SR STA
R PILOT": RETURN
2900  IF R < 116 THEN R$ = "STAR
CAPTAIN": RETURN
2910  IF R < 136 THEN R$ = "STAR
ADMIRAL": RETURN
2920 R$ = "STAR COMMANDER": RETURN

2930  RESTORE : FOR X = 1 TO 16: READ
P,D: POKE 806,P: POKE 801,D:
 CALL 800: NEXT : RETURN
2940 F5 = 1: GOSUB 1160:X = 132:S
 = 4:I = 14:Y = 115:F = 1.2:
POKE 806,10: POKE 801,60: HCOLOR= 3
2950  CALL 800:S = S * F: IF S >
Y THEN 2670
2960 SH = 9:A = X:B = Y - S: HCOLOR=
3: CALL 800:SC = 1: GOSUB 13
0: CALL 800: HCOLOR= 0: GOSUB
130: CALL 800: POKE 806,10 -
S / I: GOTO 2950
2970 ER =  PEEK (222):EL =  PEEK
(218) + PEEK (219) * 256: IF
OL = 1 THEN 1160
2980  GOTO 1080
2990  TEXT : HOME : INVERSE :BL$ =
"": VTAB 5: HTAB (14): PRINT
"O B E L I S K": NORMAL : PRINT
: PRINT  TAB( 12)"BY COMPUH
OUSE  "
3000  VTAB 10: GOSUB 3050: PRINT
: PRINT  TAB( 4)"DESTROY THE
ALIEN OBELISKS, FIND": PRINT
 TAB( 4)"YOUR BASE SHIP, AND
 BLAST OFF!!!": PRINT : GOSUB
3050
3010  PRINT : PRINT : PRINT  TAB(
14)"COPYRIGHT 1985": PRINT  TAB(
12)"BY   COMPUHOUSE    "
3020  VTAB 1: HTAB 1: FOR X = 1 TO
39:BL$ = BL$ + " ": NEXT : INVERSE
: PRINT BL$: FOR X = 2 TO 23: PRINT
" ";: HTAB 39: PRINT " ": NEXT
: PRINT BL$;: NORMAL
3030  VTAB 20: IF  NOT NX THEN  GOSUB
 2930: HTAB (8): FLASH : PRINT
" <<HIT ANY KEY TO BEGIN>>";
: NORMAL : GET A$: PRINT : VTAB
20: HTAB (8): PRINT "USE <ES
C> FOR INSTRUCTIONS";
3040  VTAB 12: HTAB 2: PRINT  LEFT$
(BL$,35);: HTAB 15: FLASH : PRINT
"<< BATTLE >>";: NORMAL : VTAB
13: HTAB 2: PRINT  LEFT$ (BL
$,35);: HTAB 15: FLASH : PRINT
"<< ALERT! >>";: NORMAL : GOTO
1080
3050  HTAB 4: FOR X = 1 TO 32: PRINT
"-";: NEXT : PRINT : RETURN

3060  DATA 18,250,24,250,23,75,22
,72,19,75,28,250,24,250,23,7
5,22,75,19,75,28,250,24,250,
23,100,22,100,23,100,19,250:
 DATA 162,48,141,48,192,169,
16,32,168,252,202,208,245,96
,162,1,141,48,192,138,105,56
,32,168,252,232,224,21,208,2
42,96
3070  DATA 18,250,24,250,23,75,22
,72,19,75,28,250,24,250,23,7
5,22,75,19,75,28,250,24,250,
23,100,22,100,23,100,19,250
```

用GRAPHICS MAGAICIAN編寫的遊戲

麥法基

RAIN SHOWER

RAIN SHOWER是一個很有趣味性的遊戲，畫面上有三層衣服，正在讓可愛的陽光把它們晒乾。可是天有不測之風雲，忽然有雨點落下來，玩者要設法移動該三層衣服，不要讓它們給雨點打中，而玩者有三次機會進行這個遊戲。

遊戲進行以JOYSTICK控制人的上下移動，PDL(0)是垂直的控制，而PDL(1)是控制三層衣服的水平移動。雨點以不規則的位置從天空落下來，雨點的數目會隨着難度相繼增加，而且到達某一個積分雨點速度亦會加快。

這個程式並非很長，所以編寫方面亦簡單而比較容易，LISTING#1是程式原始碼，裡面筆者比較詳細列出一些重要的附註讓讀者對程式編寫方面有所瞭解。

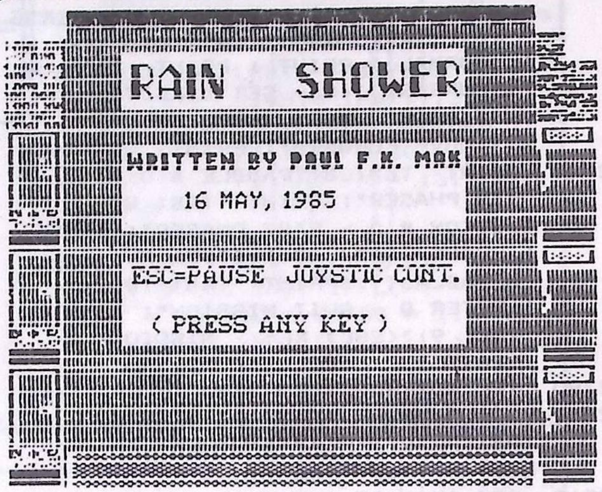
遊戲以GRAPHIC MAGICIAN作爲動畫部份，所以物體移動都交由幾條PATH負責。遊戲內人物和衣服的PATH都只有一個移動速度值(PATH VALUE)，因爲速度快慢只有一個。所改變的只是方向而已，方向值則由PDL(0)和PDL(1)決定。(請參考399和471行部份程式)

至於雨點降落的速度卻隨着難度而不同，倘若以一個固定速度降落，則遊戲難度會十分高。所以筆者在每一條雨點的PATH內分別包括有多個數值。$8EC9開始是其中一條PATH，該PATH由7個數值組成，第一個數值$20是下降速度，第二個至第六個$FF是停止移動數值，而第七個$00表示PATH完結。這樣雨點下降時會首先向下移動兩個單位，然後停止移動6次才繼續移動。換言之，雨點每下降一步便要停止6步的時間，這樣在遊戲進行時雨點移動會減慢達到所需效果。

假如遊戲積分達到一定數值時，只要把該PATH的$FF值更改爲$20。每改一個數值則雨點速度會加快一個步數的時間，於是遊戲的難度便會跟着增加了。

不要少看這個遊戲，當所有PATH值都改爲$20時，會令你有手足無措的感覺。所以遊戲開始時玩者可以選擇較高難度進行，如自命是遊戲高手的話，請試試ADVANCE這個選擇吧。

遊戲有兩個版本可供選擇，如果有MOCKING-BOARD在第4 SLOT的話，可以在遊戲進行時配



上背景音樂。電腦時代的磁碟會自動找尋是否第4 SLOT上有MOCKINGBOARD，不過要留意的是它只會分辨第4 SLOT是否有CARD存在，如果沒有MOCKINGBOARD，請不要在第4 SLOT上插上任何附件，否則遊戲不會產生聲音。

這裡有一個有趣的問題是第4 SLOT通常都會插上一張Z80用以運行CP/M軟件，倘若要和MOCKINGBOARD共存的話，請把Z80插在第2 SLOT上，因爲差不多所有爲MOCKINGBOARD而設的軟件都以第4 SLOT爲主要I/O。原則上，Z80除了第6 SLOT外都不會有問題。

本遊戲一共有多個副程式，這裡只列出其中一個主程式供大家參考研究。其餘副程式包括下列各FILE全部收錄在第十八期的程式磁碟之上。

```
RAIN SHOWER HELLO
RAIN.ANM
RAIN.PIC
P2L
RAIN SHOWER1
RAIN SHOWER
RAIN.OB
LOVE IS BLUE
TITLE.PIC
MUSIC
SLOT FINDER
```

下次再編寫另外一個遊戲，再會！

```
:A
 923   923 '
   \
:ASM
                ************************************
                *                                  *
                *  RAIN SHOWER SOURCE CODE         *
                *  BY PAUL F.K. MAK                *
                *  16 MAY, 1985                    *
                *                                  *
                ************************************
                *
                          ORG  $6000
6000: 4C 39 63            JMP  BEGIN
                *
                *
                *
                *
                *
                XBYTE     EQU  $0          ;BYTE POSITION FOR DRAW
                XBIT      EQU  $1          ;BIT POSTION FOR DRAW
                VERT      EQU  $2          ;VERT POSTION FOR DRAW
                SHNUM     EQU  $3          ;SHAPE NO. FOR DRAW
                MOVEFLAG EQU   $D0
                COLLISIONEQU   $D1         ;COLLISION FLAG
                RPITCH    EQU  $D2         ;PITCH OF SOUND WHEN MAN IS MOVING
                HIRESL    EQU  $D3         ;PAGE1 GRAPHIC LOW POINTER
                HIRESH    EQU  $D4         ;PAGE1 GRAPHIC HI POINTER
                HIRES2L   EQU  $D5         ;PAGE2 GRAPHIC LOW POINTER
                HIRES2H   EQU  $D6         ;PAGE2 GRAPHIC HI POINTER
                SHFTREGL EQU   $E0         ;SHIFT VRIABLE FOR COLLISION SOUND
                SHFTREGH EQU   $E1
                TIMEL     EQU  $E2         ;LENGTH OF BURST
                TIMEH     EQU  $E3
                FILTER    EQU  $E4         ;CUT OFF SOUND
                DELTA     EQU  $E5         ;CHANGE RATE (SOUND)
                RXTEMP    EQU  $E8         ;TEMP VARIABLE TO HOLD RAIN BYTE R
EGISTER-X
                RAINNO    EQU  $E9         ;NUMBERS OF RAIN IN AIR
                MVFLAG    EQU  $FB         ;MAN VERITICAL FLAG
                ZERO      EQU  $FC         ;PDL(0) TEMP VARIALBE
                ONE       EQU  $FD         ;PDL(1) TEMP VARIABLE
                TIME      EQU  $FE         ;TIME OF SOUND
                PITCH     EQU  $FF         ;SOUND PITCH
                *
                *
                *
                *
                C1XBYTE   EQU  $94A0       ;CLOTH BYTE POSITION
                C1XBIT    EQU  $94A1       ;CLOTH1 BIT POSITION
                C1VERT    EQU  $94A2       ;CLOTH1 VERT POSITION
                CSHNUM    EQU  $9400       ;CLOTH SHAPE NO.
                C2XBYTE   EQU  $94A9       ;CLOTH2 BYTE POSITION
                C2XBIT    EQU  $94AA       ;CLOTH2 BIT POSITION
                C2VERT    EQU  $94AB       ;CLOTH2 VERT POSITION
                C3XBYTE   EQU  $94B2       ;CLOTH3 BYTE POSITION
                C3XBIT    EQU  $94B3       ;CLOTH3 BIT POSITION
                C3VERT    EQU  $94B4       ;CLOTH3 VERT POSITION
                C1PATH    EQU  $8EFE       ;CLOTH1 PATH
                C2PATH    EQU  $8EFC       ;CLOTH2 PATH
                C3PATH    EQU  $8EFA       ;CLOTH3 PATH
                RXBYTE    EQU  $94BB       ;RAIN XBYTE
                RXBIT     EQU  $94BC       ;RAIN XBIT
                RVERT     EQU  $94BD       ;RAIN VERT
                RSHNUM    EQU  $9409       ;RAIN SHAPE NO.
                RPATH1    EQU  $945C       ;PATH LOCATION OF RAIN1 FOR SOUND
                YVERTL    EQU  $92C0       ;LOOK-UP TABLE FOR PAGE 1 GRAPHIC
                YVERTH    EQU  $9200
                YVERT2L   EQU  $67C0       ;LOOK-UP TABLE FOR PAGE 2 GRAPHIC
                YVERT2H   EQU  $6700
                SPEAKER   EQU  $C030
                BUTTON    EQU  $C061
                KEY       EQU  $C000       ;KEY PRESS
                RESETKEY EQU   $C010       ;RESET KEY I/O
                *
                SCORE0    DS   1           ;SCORE VALUE VARIALBES
                SCORE1    DS   1
                SCORE2    DS   1
                SCORE3    DS   1
                HSCORE0   DS   1           ;HIGH SCORE VALE VARIALBES
                HSCORE1   DS   1
                HSCORE2   DS   1
                HSCORE3   DS   1
                *
                * PRINT SCORE ROUTINE *
                *
600B: A9 00     PRINT     LDA  #$0         ;X & Y COORDINATES OF 1ST SCORE
600D: 85 00               STA  XBYTE
600F: 85 01               STA  XBIT
6011: 85 02               STA  VERT
6013: AD 06 60            LDA  SCORE3      ;SCORE VALUE
6016: 18                  CLC
6017: 69 03               ADC  #$3         ;ADD SHAPE NUMBER OFF-SET
6019: 85 03               STA  SHNUM       ;STORE TO SHAPE NUMBER
601B: 20 EB 63            JSR  XDRAW       ;DRAW ON SCREEN
601E: A9 01               LDA  #$1
6020: 85 00               STA  XBYTE
6022: AD 05 60            LDA  SCORE2
6025: 18                  CLC
6026: 69 03               ADC  #$3
6028: 85 03               STA  SHNUM
602A: 20 EB 63            JSR  XDRAW
602D: A9 02               LDA  #$2
602F: 85 00               STA  XBYTE
6031: AD 04 60            LDA  SCORE1
6034: 18                  CLC
6035: 69 03               ADC  #$3
6037: 85 03               STA  SHNUM
6039: 20 EB 63            JSR  XDRAW
603C: A9 03               LDA  #$3
603E: 85 00               STA  XBYTE
6040: AD 03 60            LDA  SCORE0
6043: 18                  CLC
6044: 69 03               ADC  #$3
6046: 85 03               STA  SHNUM
6048: 20 EB 63            JSR  XDRAW
604B: 60                  RTS
                *
                ** HIGH SCORE ROUTINE **
                *
                *
604C: A9 24     HPRINT    LDA  #$24
604E: 85 00               STA  XBYTE
6050: A9 00               LDA  #$0
6052: 85 01               STA  XBIT
6054: 85 02               STA  VERT
6056: AD 0A 60            LDA  HSCORE3
6059: 18                  CLC
605A: 69 03               ADC  #$3
605C: 85 03               STA  SHNUM
605E: 20 EB 63            JSR  XDRAW
6061: A9 25               LDA  #$25
6063: 85 00               STA  XBYTE
6065: AD 09 60            LDA  HSCORE2
6068: 18                  CLC
6069: 69 03               ADC  #$3
606B: 85 03               STA  SHNUM
606D: 20 EB 63            JSR  XDRAW
6070: A9 26               LDA  #$26
6072: 85 00               STA  XBYTE
6074: AD 08 60            LDA  HSCORE1
6077: 18                  CLC
6078: 69 03               ADC  #$3
607A: 85 03               STA  SHNUM
607C: 20 EB 63            JSR  XDRAW
607F: A9 27               LDA  #$27
6081: 85 00               STA  XBYTE
6083: AD 07 60            LDA  HSCORE0
6086: 18                  CLC
6087: 69 03               ADC  #$3
6089: 85 03               STA  SHNUM
608B: 20 EB 63            JSR  XDRAW
608E: 60                  RTS
                *
                ** SCORE COUNING ROUTINE **
                *
608F: EE 03 60  SCORE     INC  SCORE0      ;INCREMENT COUNTER
6092: AD 03 60            LDA  SCORE0
6095: C9 0A               CMP  #$A         ;IF <10 DON'T CARRY TENS DIGIT
6097: 90 41               BLT  ENSCORE
6099: A9 00               LDA  #$0         ;ZERO OUT 1'S DIGIT
609B: 8D 03 60            STA  SCORE0
609E: EE 04 60            INC  SCORE1      ;ADD CARRY IN TENS
60A1: 20 93 63            JSR  ADDONE
60A4: AD 04 60            LDA  SCORE1
60A7: C9 0A               CMP  #$A
60A9: 90 2F               BLT  ENSCORE
60AB: A9 00               LDA  #$0
60AD: 8D 04 60            STA  SCORE1
60B0: EE 05 60            INC  SCORE2
60B3: 20 B2 65            JSR  DIFFICULTY ;IF SCORE>100 THEN CHANGE DIFFICUL
TY
60B6: AD 05 60            LDA  SCORE2
60B9: C9 0A               CMP  #$A
60BB: 90 1D               BLT  ENSCORE
60BD: A9 00               LDA  #$0
60BF: 8D 05 60            STA  SCORE2
60C2: EE 06 60            INC  SCORE3
60C5: AD 06 60            LDA  SCORE3
60C8: C9 0A               CMP  #$A
60CA: 90 0E               BLT  ENSCORE
60CC: A9 00               LDA  #$0
60CE: 8D 03 60            STA  SCORE0
60D1: 8D 04 60            STA  SCORE1
60D4: 8D 05 60            STA  SCORE2
60D7: 8D 06 60            STA  SCORE3
60DA: 60        ENSCORE   RTS
                *
                *
                *
                MANVERT   DS   1
                *
                *
60DC: AD 30 C0  SOUND     LDA  SPEAKER     ;SOUND ROUTINE
60DF: 88        S1        DEY
60E0: D0 04               BNE  S2
60E2: C6 FE               DEC  TIME
60E4: F0 08               BEQ  SEND
60E6: CA        S2        DEX
60E7: D0 F6               BNE  S1
60E9: A6 FF               LDX  PITCH
60EB: 4C DC 60            JMP  SOUND
60EE: 60        SEND      RTS
                *
                *  WAIT ROUTINE FOR SMOTHING SOUND *
                *
                *
60EF: AD 00 70  WAIT      LDA  $7000
60F2: 88        W1        DEY
60F3: D0 04               BNE  W2
60F5: C6 FE               DEC  TIME
60F7: F0 08               BEQ  WEND
60F9: CA        W2        DEX
60FA: D0 F6               BNE  W1
60FC: A6 FF               LDX  PITCH
60FE: 4C EF 60            JMP  WAIT
6101: 60        WEND      RTS
                *
                *  SOUND OF COLLISION *
                *
6102: 38        CSOUND    SEC
6103: E6 E3               INC  TIMEH       ;MAKES THE LOOP RIGHT
6105: A6 E4               LDX  FILTER      ;ASSUME FIRST CLICK
6107: 26 E0     CLOOP1    ROL  SHFTREGL
6109: 26 E1               ROL  SHFTREGH    ;SHIFT A BIT OUT
610B: 8A                  TXA              ;CHECK FOR CUTOFF PERIOD
610C: F0 01               BEQ  PASS        ;NEVER NEGATIVE
610E: CA                  DEX
610F: D0 07     PASS      BNE  NOCLICK     ;PERIOD TOO SHORT
6111: 90 05               BCC  NOCLICK     ;DIDN'T WANT TO ANYWAY
6113: AD 30 C0  CLICK     LDA  SPEAKER
6116: A6 E4               LDX  FILTER      ;DON'T CLICK AGAIN TILL TIMEOUT
6118: 6A        NOCLICK   ROR              ;ALIGN CARRY AND
6119: 6A                  ROR              ;BIT #14 OF SHFTREG
611A: 6A                  ROR
611B: 45 E1               EOR  SHFTREGH    ;X-OR THOSE BITS
611D: 0A                  ASL
611E: 0A                  ASL
611F: 0A                  ASL              ;PUT BACK IN CARRY
6120: 08        SWEEP     PHP              ;SAVE CARRY ON STACK
6121: A5 E5               LDA  DELTA
6123: F0 14               BEQ  NODELTA
6125: 88                  DEY
6126: D0 11               BNE  NODELTA
6128: A8                  TAY
6129: 30 09               BMI  DOWNIT
612B: A5 E4     UPIT      LDA  FILTER
612D: F0 0A               BEQ  NODELTA     ;NO CHANGE NEEDED
612F: C6 E4               DEC  FILTER
6131: 98                  TYA
6132: D0 05               BNE  NODELTA     ;NOT TIME FOR CHANGE
6134: E6 E4     DOWNIT    INC  FILTER      ;SWEEP DOWN
```

```
6136: 29 7F              AND   #$7F
6138: A8                 TAY               ;AND RESET COUNTER
6139: 28        NODELTA  PLP               ;RESTORE CARRY
613A: C6 E2              DEC   TIMEL
613C: D0 C9              BNE   CLOOP1
613E: C6 E3              DEC   TIMEH
6140: D0 C5              BNE   CLOOP1
6142: 60                 RTS
                *
                RARRY    DS    10
                *
                *
                * ROUTINE TO TRANSFER PAGE 2 GRAPHIC TO PAGE 1 *
                *
614D: A2 00     PAGE     LDX   #$00        ;TOP ROW OF SCREEN
614F: 20 83 65  PAGE1    JSR   GETADR      ;GET ADDRESS FROM PAGE 1
6152: 20 8E 65           JSR   GETADR1     ;GET ADDRESS FROM PAGE 2
6155: A0 00              LDY   #$00
6157: B1 D5     PAGE2    LDA   (HIRES2L),Y ;HORIZ ADDRESS VALUE
6159: 91 D3              STA   (HIRESL),Y  ;COPY TO PAGE 2
615B: C8                 INY               ;INCREASE HORIZ ADDRESS BY ONE
615C: C0 28              CPY   #$28        ;HORIZ ADDRESS FINISHED?
615E: 90 F7              BLT   PAGE2       ;NO, COPY HORIZ AGAIN
6160: E8                 INX               ;YES , NEXT ROW
6161: E0 BF              CPX   #$BF        ;ALL ROWS FINISHED?
6163: 90 EA              BLT   PAGE1       ;NO, COPY NEXT ROW AGAIN
6165: 60                 RTS               ;YES, DONE
                *
                *
                ** SPEED OF RAINS IN DIFFERENT LEVEL **
                *
6166: A2 00     ADVANCE1 LDX   #$00
6168: A0 00              LDY   #$00
616A: A9 00              LDA   #$00
616C: 9D CE 8E  ALOOP1   STA   $8ECE,X     ;STORE ZERO TO PATH TO SLOW DOWN R
AIN
616F: E8                 INX
6170: E8                 INX
6171: E8                 INX
6172: E8                 INX
6173: E8                 INX
6174: E8                 INX
6175: E8                 INX
6176: C8                 INY
6177: C0 07              CPY   #$7
6179: 90 F1              BLT   ALOOP1
617B: 60                 RTS
                *
617C: A2 00     ADVANCE2 LDX   #$00
617E: A0 00              LDY   #$00
6180: A9 00              LDA   #$00
6182: 9D CD 8E  ALOOP2   STA   $8ECD,X
6185: E8                 INX
6186: E8                 INX
6187: E8                 INX
6188: E8                 INX
6189: E8                 INX
618A: E8                 INX
618B: E8                 INX
618C: C8                 INY
618D: C0 07              CPY   #$7
618F: 90 F1              BLT   ALOOP2
6191: 60                 RTS
                *
6192: A2 00     ADVANCE3 LDX   #$00
6194: A0 00              LDY   #$00
6196: A9 00              LDA   #$00
6198: 9D CC 8E  ALOOP3   STA   $8ECC,X
619B: E8                 INX
619C: E8                 INX
619D: E8                 INX
619E: E8                 INX
619F: E8                 INX
61A0: E8                 INX
61A1: E8                 INX
61A2: C8                 INY
61A3: C0 07              CPY   #$7
61A5: 90 F1              BLT   ALOOP3
61A7: 60                 RTS
                *
61A8: A2 00     ADVANCE4 LDX   #$00
61AA: A0 00              LDY   #$00
61AC: A9 00              LDA   #$00
61AE: 9D CB 8E  ALOOP4   STA   $8ECB,X
61B1: E8                 INX
61B2: E8                 INX
61B3: E8                 INX
61B4: E8                 INX
61B5: E8                 INX
61B6: E8                 INX
61B7: E8                 INX
61B8: C8                 INY
61B9: C0 07              CPY   #$7
61BB: 90 F1              BLT   ALOOP4
61BD: 60                 RTS
                *
                *
                *
                *
                *
                * ROUTINE TO DRAW "GAME OVER" *
                *
                *
61BE: A9 0A     OVER     LDA   #$A
61C0: 85 00              STA   XBYTE
61C2: A9 00              LDA   #$0
61C4: 85 01              STA   XBIT
61C6: A9 50              LDA   #$50
61C8: 85 02              STA   VERT
61CA: A9 0D              LDA   #$D
61CC: 85 03              STA   SHNUM
61CE: 20 EB 63           JSR   XDRAW
61D1: A9 14              LDA   #$14
61D3: 85 00              STA   XBYTE
61D5: A9 0E              LDA   #$E
61D7: 85 03              STA   SHNUM
61D9: 20 EB 63           JSR   XDRAW
61DC: 60                 RTS
                *
                *
                *
61DD: A2 00     READPDL  LDX   #$00        ;PADDLE READ ROUTINE
61DF: 86 FD              STX   ONE
61E1: 86 FC              STX   ZERO
61E3: A2 7F              LDX   #$7F
61E5: AD 70 C0           LDA   $C070       ;BOTH TIMERS
61E8: AD 64 C0  RLOOP    LDA   $C064       ;PADDLE 0 TIMER
61EB: 29 80              AND   #$80
61ED: 0A                 ASL
61EE: 2A                 ROL
61EF: 65 FC              ADC   ZERO
61F1: 85 FC              STA   ZERO
61F3: AD 65 C0           LDA   $C065       ;PADDLE 1 TIMER
61F6: 29 80              AND   #$80
61F8: 0A                 ASL
61F9: 2A                 ROL
61FA: 65 FD              ADC   ONE
61FC: 85 FD              STA   ONE
61FE: CA                 DEX
61FF: D0 E7              BNE   RLOOP
6201: A9 7F              LDA   #$7F
6203: 38                 SEC
6204: E5 FC              SBC   ZERO
6206: 85 FC              STA   ZERO
6208: A9 7F              LDA   #$7F
620A: 38                 SEC
620B: E5 FD              SBC   ONE
620D: 85 FD              STA   ONE
620F: 60                 RTS
                *
                *
                *     MOVE CLOTHS ROUTINE
                *
6210: A5 FD     HMOVE    LDA   ONE         ;DON'T READ PDL(1)
6212: C9 5A              CMP   #$5A
6214: 90 0E              BLT   MOVEA
6216: C9 78              CMP   #$78
6218: B0 0A              BGE   MOVEA
621A: A5 FC     MOVE1    LDA   ZERO        ;READ PDL(0)
621C: C9 5A              CMP   #$5A
621E: 90 10              BLT   MOVE2
6220: C9 78              CMP   #$78
6222: B0 43              BGE   MOVE3
6224: A9 FF     MOVEA    LDA   #$FF
6226: 8D FE 8E           STA   C1PATH      ;DONE'T MOVE
6229: 8D FC 8E           STA   C2PATH
622C: 8D FA 8E           STA   C3PATH
622F: 60                 RTS
6230: A5 FB     MOVE2    LDA   MVFLAG      ;LOCATE VALUE OF MVFLAG
6232: C9 00              CMP   #$0         ;=0,MAN ON 1ST COMPARTMENT
6234: F0 13              BEQ   MOVE2A      ;
6236: C9 01              CMP   #$1         ;=1,MAN ON 2ND COMPARTMENT
6238: F0 1E              BEQ   MOVE2B      ;
623A: AD B2 94           LDA   C3XBYTE     ;CHECK CLOTH3 BYTE POSTION
623D: C9 10              CMP   #$10        ;RIGHT LIMIT
623F: F0 E3              BEQ   MOVEA       ;DON'T MOVE
6241: A9 03              LDA   #$03
6243: 8D FA 8E           STA   C3PATH      ;MOVE CLOTH3 2 UNITS RIGHT
6246: 4C 9B 62           JMP   MOVE3C      ;RETURN
6249: AD A0 94  MOVE2A   LDA   C1XBYTE     ;CHECK CLOTH1 BYTE POSTION
624C: C9 10              CMP   #$10
624E: F0 D4              BEQ   MOVEA
6250: A9 03              LDA   #$03
6252: 8D FE 8E           STA   C1PATH      ;MOVE CLOTH1 2 UNITS RIGHT
6255: 4C 9B 62           JMP   MOVE3C      ;RETURN
6258: AD A9 94  MOVE2B   LDA   C2XBYTE     ;CHECK CLOTH2 BYTE POSTION
625B: C9 10              CMP   #$10
625D: F0 C5              BEQ   MOVEA
625F: A9 03              LDA   #$03
6261: 8D FC 8E           STA   C2PATH      ;MOVE CLOTH2 2 UNITS RIGHT
6264: 4C 9B 62           JMP   MOVE3C      ;RETURN
6267: A5 FB     MOVE3    LDA   MVFLAG      ;CHECK MVFLAG VALUE
6269: C9 00              CMP   #$00        ;=0,MAN ON 1ST COMPARTMENT
626B: F0 13              BEQ   MOVE3A      ;
626D: C9 01              CMP   #$1         ;=1,MAN ON 2ND COMPARTMENT
626F: F0 1E              BEQ   MOVE3B      ;MOVE LEFT
6271: AD B2 94           LDA   C3XBYTE     ;CHECK CLOTH3 BYTE POSTN
6274: C9 04              CMP   #$4         ;LEFT LIMIT
6276: F0 2C              BEQ   MOVE3D      ;REACH LIMIT,DON'T MOVE
6278: A9 0C              LDA   #$0C
627A: 8D FA 8E           STA   C3PATH      ;MOVE CLOTH3 2 UNITS LEFT
627D: 4C 9B 62           JMP   MOVE3C
6280: AD A0 94  MOVE3A   LDA   C1XBYTE     ;CHECK CLOTH1 BYTE POSTN
6283: C9 04              CMP   #$4
6285: F0 1D              BEQ   MOVE3D      ;REACH LIMIT,DON'T MOVE
6287: A9 0C              LDA   #$0C
6289: 8D FE 8E           STA   C1PATH      ;MOVE CLOTH1 2 UNITS LEFT
628C: 4C 9B 62           JMP   MOVE3C
628F: AD A9 94  MOVE3B   LDA   C2XBYTE     ;CHECK CLOTH2 BYTE POSTN
6292: C9 04              CMP   #$4
6294: F0 0E              BEQ   MOVE3D      ;REACH LIMIT,DON'T MOVE
6296: A9 0C              LDA   #$0C
6298: 8D FC 8E           STA   C2PATH      ;MOVE CLOTH2 2 UNITS LEFT
629B: A0 01     MOVE3C   LDY   #$01        ;SOUND OF PULLING CLOTHS
629D: 8D 30 C0  SOUNDO   STA   SPEAKER
62A0: 88                 DEY
62A1: D0 FA              BNE   SOUNDO
62A3: 60                 RTS
62A4: 4C 24 62  MOVE3D   JMP   MOVEA
                *
                *
                *
                *
                *     MOVE MAN IN VERTICAL DIRECTION
                *
62A7: A5 FC     VMOVE    LDA   ZERO        ;DON'T READ PDL(0)
62A9: C9 5A              CMP   #$5A
62AB: 90 12              BLT   MOVEB
62AD: C9 78              CMP   #$78
62AF: B0 0E              BGE   MOVEB
62B1: A5 FD     MOVE4    LDA   ONE         ;READ PDL(1)
62B3: C9 5A              CMP   #$5A
62B5: 90 09              BLT   MOVE5
62B7: C9 78              CMP   #$78
62B9: B0 28              BGE   MOVE6
62BB: A9 00              LDA   #$0
62BD: 85 D0              STA   MOVEFLAG
62BF: 60        MOVEB    RTS
62C0: A5 D0     MOVE5    LDA   MOVEFLAG
62C2: D0 FB              BNE   MOVEB
62C4: AD DB 60           LDA   MANVERT     ;MAN VERTICAL POSTION
62C7: C9 9A              CMP   #$9A        ;LOWER LIMIT
62C9: F0 F4              BEQ   MOVEB       ;EQUAL TO LOWER LIMIT,NO MOVE
62CB: 20 24 63           JSR   XDRAWMAN    ;XDRAW MAN IN PRESENT POSTN
62CE: AD DB 60           LDA   MANVERT     ;MAN VERTICAL POSITION
62D1: 18                 CLC               ;
62D2: 69 30              ADC   #$30        ;ADD OFFSET
62D4: 8D DB 60           STA   MANVERT
62D7: 20 0F 63           JSR   DRAWMAN
62DA: A9 01              LDA   #$1
62DC: 85 D0              STA   MOVEFLAG
62DE: E6 FB              INC   MVFLAG      ;INCREASE MVFLAG BY ONE
```

```
62E0: 4C 03 63            JMP   MOVE7
62E3: A5 D0     MOVE6     LDA   MOVEFLAG
62E5: D0 DB               BNE   MOVEB
62E7: AD DB 60            LDA   MANVERT
62EA: C9 3A               CMP   #$3A       ;UPPER LIMIT
62EC: F0 D1               BEQ   MOVEB      ;EQUAL TO UPPER LIMIT,NO MOVE
62EE: 20 24 63            JSR   XDRAWMAN   ;XDRAW MAN IN PRESENT POSTN
62F1: AD DB 60            LDA   MANVERT
62F4: 38                  SEC              ;
62F5: E9 30               SBC   #$30
62F7: 8D DB 60            STA   MANVERT
62FA: 20 0F 63            JSR   DRAWMAN    ;DRAW MAN IN UPPER POSTN
62FD: A9 01               LDA   #$1
62FF: 85 D0               STA   MOVEFLAG
6301: C6 FB               DEC   MVFLAG     ;DECREASE MVFLAG BY ONE
6303: A9 1F     MOVE7     LDA   #$1F       ;G NOTE FOR SOUND OF MOVING MAN
6305: 85 FF               STA   PITCH
6307: A9 04               LDA   #$4
6309: 85 FE               STA   TIME
630B: 20 DC 60            JSR   SOUND
630E: 60                  RTS
                *
                *
                *
630F: A9 24     DRAWMAN   LDA   #$24
6311: 85 00               STA   XBYTE
6313: A9 00               LDA   #$0
6315: 85 01               STA   XBIT
6317: AD DB 60            LDA   MANVERT
631A: 85 02               STA   VERT
631C: A9 00               LDA   #$0
631E: 85 03               STA   SHNUM
6320: 20 00 8F            JSR   $8F00
6323: 60                  RTS
                *
                *
                *
6324: A9 24     XDRAWMAN  LDA   #$24
6326: 85 00               STA   XBYTE
6328: A9 00               LDA   #$0
632A: 85 01               STA   XBIT
632C: AD DB 60            LDA   MANVERT
632F: 85 02               STA   VERT
6331: A9 00               LDA   #$0
6333: 85 03               STA   SHNUM
6335: 20 A2 8F            JSR   $8FA2
6338: 60                  RTS
                *
                *
                *
6339: A9 3A     BEGIN     LDA   #$3A       ;INITIAL VERT POSTN OF MAN
633B: 8D DB 60            STA   MANVERT
633E: 20 0F 63            JSR   DRAWMAN
6341: A9 00               LDA   #$0
6343: 85 FB               STA   MVFLAG     ;MVFLAG=0
6345: 85 D0               STA   MOVEFLAG
6347: 85 D1               STA   COLLISION
6349: A9 2F               LDA   #$2F
634B: 85 D2               STA   RPITCH     ;SOUND OF C NOTE
                *
                *
                *
                *   MAIN PROGRAM LOOP
                *
634D: 20 DD 61  START     JSR   READPDL    ;READ PADDLES
6350: 20 A7 62            JSR   VMOVE      ;MOVE MAN
6353: 20 10 62            JSR   HMOVE      ;MOVE CLOTHS
6356: 20 C1 63            JSR   CHKR       ;CHECK IF RAIN REACHES BOTTOM
6359: 20 9C 63            JSR   ADDONE1    ;HORIZ POSTN OF RAIN
635C: 20 99 65            JSR   PAUSE      ;IF ESC PRESSED, THEN STOP
635F: 20 75 64            JSR   RCOLLIDE1
6362: 20 CF 64            JSR   RCOLLIDE2
6365: 20 29 65            JSR   RCOLLIDE3
6368: A5 D1               LDA   COLLISION
636A: D0 26               BNE   RETURN
636C: 20 40 90            JSR   $9040      ;CALL ANIMATION ROUTINE
636F: AD 5C 94            LDA   RPATH1
6372: C9 F4               CMP   #$F4
6374: D0 0E               BNE   LOOP1
6376: A5 D2               LDA   RPITCH
6378: 85 FF               STA   PITCH
637A: A9 02               LDA   #$2
637C: 85 FE               STA   TIME
637E: 20 DC 60            JSR   SOUND
6381: 4C 8F 63            JMP   LOOP2
6384: A5 D2     LOOP1     LDA   RPITCH
6386: 85 FF               STA   PITCH
6388: A9 02               LDA   #$2
638A: 85 FE               STA   TIME
638C: 20 EF 60            JSR   WAIT
638F: 4C 4D 63  LOOP2     JMP   START      ;LOOP BACK
6392: 60        RETURN    RTS              ;RETURN TO BASIC
                *
                *
                *
                *
                *
                * TIME TO ADD MORE RAIN ROUTINE
                *
6393: A5 E9     ADDONE    LDA   RAINNO
6395: C9 06               CMP   #$6
6397: B0 02               BGE   DOMORE
6399: E6 E9               INC   RAINNO
639B: 60        DOMORE    RTS
                *
                *
                *
                * INITIAL POSTIONS OF ALL RAINS WHEN MAX NOS. OF
                * RAINS IN AIR
                *
639C: A2 00     ADDONE1   LDX   #$00       ;INDEX=0
639E: A0 00               LDY   #$00       ;Y-INDEX=0
63A0: BD BD 94  ADDONE2   LDA   RVERT,X    ;CHECK RAIN'S VERT POTSN
63A3: C9 00               CMP   #$0        ;=0?
63A5: F0 0B               BEQ   ADDONE4    ;YES, PUT RAIN ONTO SCREEN
63A7: E8        ADDONE3   INX              ;NO, CHECK ANOTHER ONE
63A8: E8                  INX
63A9: E8                  INX
63AA: C8                  INY
63AB: C4 E9               CPY   RAINNO     ;CHECK ALL RAINS?
63AD: 90 F1               BLT   ADDONE2    ;NO, CHECK AGAIN
63AF: 4C C0 63            JMP   ADDONE5    ;RETURN
63B2: BD A0 94  ADDONE4   LDA   C1XBYTE,X  ;HORIZ POSTN OF CLOTH
63B5: 18                  CLC
63B6: 69 01               ADC   #$1
63B8: 9D BB 94            STA   RXBYTE,X
63BB: A9 00               LDA   #$0
63BD: 99 43 61            STA   RARRY,Y    ;RESET RARRY TO ZERO FOR SCORE
63C0: 60        ADDONE5   RTS
                *
                *
                * CHECK RAIN'S VERT POSITION FOR ADDING SCORE *
                *
63C1: A2 00     CHKR      LDX   #$00
63C3: A0 00               LDY   #$00
63C5: BD BD 94  CHKR1     LDA   RVERT,X    ;VERT POSTN OF RAIN
63C8: C9 C0               CMP   #$C0       ;>192?
63CA: B0 0B               BGE   CHKR3      ;YES, TO ADD SCORE
63CC: E8        CHKR2     INX
63CD: E8                  INX
63CE: E8                  INX
63CF: C8                  INY
63D0: C4 E9               CPY   RAINNO     ;CHECK ALL RAINS?
63D2: 90 F1               BLT   CHKR1      ;NO, CHECK ANOTHER ONE
63D4: 4C EA 63            JMP   CHKR5      ;YES, RETURN
63D7: B9 43 61  CHKR3     LDA   RARRY,Y    ;SCORE ALREADY ADDED?
63DA: D0 F0               BNE   CHKR2      ;YES, CHECK ANOTHER ONE
63DC: A9 01               LDA   #$1
63DE: 99 43 61            STA   RARRY,Y    ;NO, RESET RARRY
63E1: 20 0B 60            JSR   PRINT      ;CLEAR OLD SCORE
63E4: 20 8F 60            JSR   SCORE      ;INCREASE SCORE BY ONE
63E7: 20 0B 60            JSR   PRINT      ;PRINT NEW SCORE
63EA: 60        CHKR5     RTS              ;RETURN
                *
                *
                *
63EB: A9 00     XDRAW     LDA   #$00       ;DRAW SCORE ROUTINE
63ED: 85 08               STA   $08
63EF: A5 03               LDA   SHNUM
63F1: 0A                  ASL
63F2: A8                  TAY
63F3: B9 80 93            LDA   $9380,Y
63F6: 85 08               STA   $08
63F8: C8                  INY
63F9: B9 80 93            LDA   $9380,Y
63FC: 85 09               STA   $09
63FE: A5 01               LDA   XBIT
6400: 0A                  ASL
6401: A8                  TAY
6402: B1 08               LDA   ($08),Y
6404: AA                  TAX
6405: C8                  INY
6406: B1 08               LDA   ($08),Y
6408: 18                  CLC
6409: 65 08               ADC   $08
640B: 85 08               STA   $08
640D: 8A                  TXA
640E: 65 09               ADC   $09
6410: 85 09               STA   $09
6412: A0 01               LDY   #$01
6414: B1 08     XLOOPA    LDA   ($08),Y
6416: 99 F9 20            STA   $20F9,Y
6419: 88                  DEY
641A: 10 F8               BPL   XLOOPA
641C: A5 08               LDA   $08
641E: 8D 5D 64            STA   XLOOPC+1
6421: 8D 52 64            STA   XLOOPB+1
6424: A5 09               LDA   $09
6426: 8D 5E 64            STA   XLOOPC+2
6429: 8D 53 64            STA   XLOOPB+2
642C: A2 02               LDX   #$02
642E: A5 02               LDA   VERT
6430: 85 0A               STA   $0A
6432: AD F9 20  XLOOPG    LDA   $20F9
6435: 85 05               STA   $05
6437: A4 02               LDY   VERT
6439: C0 C0               CPY   #$C0
643B: 90 04               BCC   XLOOPD
643D: A0 80               LDY   #$80
643F: B0 0C               BCS   XLOOPE
6441: B9 C0 92  XLOOPD    LDA   $92C0,Y
6444: 85 06               STA   $06
6446: B9 00 92            LDA   $9200,Y
6449: 85 07               STA   $07
644B: A4 00               LDY   XBYTE
644D: C0 28     XLOOPE    CPY   #$28
644F: B0 12               BCS   XLOOPF
6451: BD 00 70  XLOOPB    LDA   $7000,X
6454: 31 06               AND   ($06),Y
6456: 29 7F               AND   #$7F
6458: F0 02               BEQ   XLOOPC
645A: 85 0B               STA   $0B
645C: BD 00 70  XLOOPC    LDA   $7000,X
645F: 51 06               EOR   ($06),Y
6461: 91 06               STA   ($06),Y
6463: E8        XLOOPF    INX
6464: C8                  INY
6465: C6 05               DEC   $05
6467: D0 E4               BNE   XLOOPE
6469: E6 02               INC   VERT
646B: CE FA 20            DEC   $20FA
646E: D0 C2               BNE   XLOOPG
6470: A5 0A               LDA   $0A
6472: 85 02               STA   VERT
6474: 60                  RTS
                *
                *
6475: A2 00     RCOLLIDE1LDX    #$00       ;CHECK COLLISION 1ST COMPARTMENT
6477: A0 00               LDY   #$00
6479: BD BD 94  RCOL1A    LDA   RVERT,X    ;VERT POSTN OF RAIN
647C: C9 29               CMP   #$29
647E: B0 09               BGE   RCOL1C     ;IF >41 THEN CHICK IF <57
6480: E8        RCOL1B    INX
6481: E8                  INX
6482: E8                  INX
6483: C8                  INY
6484: C4 E9               CPY   RAINNO     ;CHECK ALL?
6486: 90 F1               BLT   RCOL1A     ;NO CHECK AGAIN
6488: 60                  RTS              ;YES, RETURN
6489: C9 39     RCOL1C    CMP   #$39
648B: 90 03               BLT   RCOL1D     ;IF <57 THEN CHECK COLLISION
648D: 4C 80 64            JMP   RCOL1B     ;CHECK ANOTHER ONE
6490: BD BB 94  RCOL1D    LDA   RXBYTE,X   ;HORIZ POSTN OF RAIN TO BE CHECKED
6493: 85 E8               STA   RXTEMP     ;STORE TO TEMPORARY VARIABLE
6495: A2 00               LDX   #$00
6497: A0 00               LDY   #$00
6499: BD A0 94  RCOL1E    LDA   C1XBYTE,X  ;CLOTHS OF 1ST COMPARTMENT
649C: 38                  SEC
649D: E5 E8               SBC   RXTEMP     ;SUBTRACT BYTE POSTN OF RAIN
649F: F0 13               BEQ   RCOL1F     ;=0,COLLISDE OCCURS
64A1: A5 E8               LDA   RXTEMP     ;NO, SUBTRACT BYTE POSTN OF CLOTH
64A3: 38                  SEC
```

RAIN SHOWER

```
64A4: FD A0 94            SBC  C1XBYTE,X
64A7: C9 03               CMP  #$3       ;<3?
64A9: 90 09               BLT  RCOL1F    ;YES, COLLIDE OCCURS
64AB: E8                  INX
64AC: E8                  INX
64AD: E8                  INX
64AE: C8                  INY
64AF: C0 03               CPY  #$3       ;CHECK ALL CLOTHS?
64B1: 90 E6               BLT  RCOL1E    ;NO, CHECK ANOTHER CLOTH
64B3: 60                  RTS            ;CHECK ALL CLOTHS, RETURN
64B4: A9 01     RCOL1F    LDA  #$1
64B6: 85 D1               STA  COLLISION ;COLLISION FLAG=1
64B8: BD A0 94            LDA  C1XBYTE,X
64BB: 85 00               STA  XBYTE
64BD: BD A1 94            LDA  C1XBIT,X
64C0: 85 01               STA  XBIT
64C2: BD A2 94            LDA  C1VERT,X
64C5: 85 02               STA  VERT
64C7: A9 0F               LDA  #$F
64C9: 85 03               STA  SHNUM
64CB: 20 00 8F            JSR  $8F00
64CE: 60                  RTS
                *
                *
                *
64CF: A2 00     RCOLLIDE2LDX  #$00       ;CHECK COLLISION 2ND COMPARTMENT
64D1: A0 00               LDY  #$00
64D3: BD BD 94  RCOL2A    LDA  RVERT,X
64D6: C9 59               CMP  #$59
64D8: B0 09               BGE  RCOL2C
64DA: E8        RCOL2B    INX
64DB: E8                  INX
64DC: E8                  INX
64DD: C8                  INY
64DE: C4 E9               CPY  RAINNO
64E0: 90 F1               BLT  RCOL2A
64E2: 60                  RTS
64E3: C9 69     RCOL2C    CMP  #$69
64E5: 90 03               BLT  RCOL2D
64E7: 4C DA 64            JMP  RCOL2B
64EA: BD BB 94  RCOL2D    LDA  RXBYTE,X
64ED: 85 E8               STA  RXTEMP
64EF: A2 00               LDX  #$00
64F1: A0 00               LDY  #$00
64F3: BD A9 94  RCOL2E    LDA  C2XBYTE,X
64F6: 38                  SEC
64F7: E5 E8               SBC  RXTEMP
64F9: F0 13               BEQ  RCOL2F
64FB: A5 E8               LDA  RXTEMP
64FD: 38                  SEC
64FE: FD A9 94            SBC  C2XBYTE,X
6501: C9 03               CMP  #$3
6503: 90 09               BLT  RCOL2F
6505: E8                  INX
6506: E8                  INX
6507: E8                  INX
6508: C8                  INY
6509: C0 03               CPY  #$3
650B: 90 E6               BLT  RCOL2E
650D: 60                  RTS
650E: A9 01     RCOL2F    LDA  #$1
6510: 85 D1               STA  COLLISION
6512: BD A9 94            LDA  C2XBYTE,X
6515: 85 00               STA  XBYTE
6517: BD AA 94            LDA  C2XBIT,X
651A: 85 01               STA  XBIT
651C: BD AB 94            LDA  C2VERT,X
651F: 85 02               STA  VERT
6521: A9 0F               LDA  #$F
6523: 85 03               STA  SHNUM
6525: 20 00 8F            JSR  $8F00
6528: 60                  RTS
                *
                *
6529: A2 00     RCOLLIDE3LDX  #$00       ;CHECK COLLISION 3RD COMPARTMENT
652B: A0 00               LDY  #$00
652D: BD BD 94  RCOL3A    LDA  RVERT,X
6530: C9 89               CMP  #$89
6532: B0 09               BGE  RCOL3C
6534: E8        RCOL3B    INX
6535: E8                  INX
6536: E8                  INX
6537: C8                  INY
6538: C4 E9               CPY  RAINNO
653A: 90 F1               BLT  RCOL3A
653C: 60                  RTS
653D: C9 99     RCOL3C    CMP  #$99
653F: 90 03               BLT  RCOL3D
6541: 4C 34 65            JMP  RCOL3B
6544: BD BB 94  RCOL3D    LDA  RXBYTE,X
6547: 85 E8               STA  RXTEMP
6549: A2 00               LDX  #$00
654B: A0 00               LDY  #$00
654D: BD B2 94  RCOL3E    LDA  C3XBYTE,X
6550: 38                  SEC
6551: E5 E8               SBC  RXTEMP
6553: F0 13               BEQ  RCOL3F
6555: A5 E8               LDA  RXTEMP
6557: 38                  SEC
6558: FD B2 94            SBC  C3XBYTE,X
655B: C9 03               CMP  #$3
655D: 90 09               BLT  RCOL3F
655F: E8                  INX
6560: E8                  INX
6561: E8                  INX
6562: C8                  INY
6563: C0 03               CPY  #$3
6565: 90 E6               BLT  RCOL3E
6567: 60                  RTS
6568: A9 01     RCOL3F    LDA  #$1
656A: 85 D1               STA  COLLISION
656C: BD B2 94            LDA  C3XBYTE,X
656F: 85 00               STA  XBYTE
6571: BD B3 94            LDA  C3XBIT,X
6574: 85 01               STA  XBIT
6576: BD B4 94            LDA  C3VERT,X
6579: 85 02               STA  VERT
657B: A9 0F               LDA  #$F
657D: 85 03               STA  SHNUM
657F: 20 00 8F            JSR  $8F00
6582: 60                  RTS
                *
                *
                *
                *
                *
6583: BD C0 92  GETADR    LDA  YVERTL,X  ;FIND CORRECT ADDRESS ON PAGE 1
6586: 85 D3               STA  HIRESL
6588: BD 00 92            LDA  YVERTH,X
658B: 85 D4               STA  HIRESH
658D: 60                  RTS
                *
658E: BD C0 67  GETADR1   LDA  YVERT2L,X ;FIND CORRECT ADDRESS ON PAGE 2
6591: 85 D5               STA  HIRES2L
6593: BD 00 67            LDA  YVERT2H,X
6596: 85 D6               STA  HIRES2H
6598: 60                  RTS
                *
                *
6599: AD 00 C0  PAUSE     LDA  KEY       ;KEY PRESSED?
659C: 10 10               BPL  OUT       ;IF NOT, OUT
659E: C9 9B               CMP  #$9B      ;ESC KEY PRESSED?
65A0: D0 0C               BNE  OUT       ;NO, OUT
65A2: 2C 10 C0  STEP      BIT  RESETKEY  ;RESET KEY
65A5: AD 00 C0  READ      LDA  KEY       ;KEY PRESSED?
65A8: 10 FB               BPL  READ
65AA: C9 A0               CMP  #$A0      ;SPACE KEY?
65AC: D0 03               BNE  OUT1
65AE: 2C 10 C0  OUT       BIT  RESETKEY
65B1: 60        OUT1      RTS
                *
                *
                *
65B2: AD 05 60  DIFFICULTYLDA SCORE2     ;CHECK SCORE2 VALUE
65B5: C9 01               CMP  #$1       ;=1?
65B7: F0 0D               BEQ  D1        ;TO D1
65B9: C9 02               CMP  #$2       ;=2?
65BB: F0 0D               BEQ  D2        ;TO D2
65BD: C9 03               CMP  #$3       ;=3?
65BF: F0 0D               BEQ  D3        ;TO D3
65C1: C9 04               CMP  #$4       ;=4?
65C3: F0 0D               BEQ  D4        ;TO D4
65C5: 60                  RTS            ;HIGHEST DIFFICULTY, RETURN
65C6: 20 66 61  D1        JSR  ADVANCE1
65C9: 60                  RTS
65CA: 20 7C 61  D2        JSR  ADVANCE2
65CD: 60                  RTS
65CE: 20 92 61  D3        JSR  ADVANCE3
65D1: 60                  RTS
65D2: 20 A8 61  D4        JSR  ADVANCE4
65D5: 60                  RTS
                *
                *

--End assembly--
```

```
Symbol table - alphabetical order:

   ADDONE   =$6393      ADDONE1  =$639C      ADDONE2  =$63A0   ?  ADDONE3  =$63A7
   ADDONE4  =$63B2      ADDONE5  =$63C0      ADVANCE1=$6166       ADVANCE2=$617C
   ADVANCE3=$6192       ADVANCE4=$61A8       ALOOP1   =$616C      ALOOP2   =$6182
   ALOOP3   =$6198      ALOOP4   =$61AE      BEGIN    =$6339   ?  BUTTON   =$C061
   C1PATH   =$8EFE      C1VERT   =$94A2      C1XBIT   =$94A1      C1XBYTE  =$94A0
   C2PATH   =$8EFC      C2VERT   =$94AB      C2XBIT   =$94AA      C2XBYTE  =$94A9
   C3PATH   =$8EFA      C3VERT   =$94B4      C3XBIT   =$94B3      C3XBYTE  =$94B2
   CHKR     =$63C1      CHKR1    =$63C5      CHKR2    =$63CC      CHKR3    =$63D7
   CHKR5    =$63EA   ?  CLICK    =$6113      CLOOP1   =$6107      COLLISION=$D1
?  CSHNUM   =$9400   ?  CSOUND   =$6102      D1       =$65C6      D2       =$65CA
   D3       =$65CE      D4       =$65D2      DELTA    =$E5        DIFFICULTY=$65B2
   DOMORE   =$639B      DOWNIT   =$6134      DRAWMAN  =$630F      ENSCORE  =$60DA
   FILTER   =$E4        GETADR   =$6583      GETADR1  =$658E      HIRES2H  =$D6
   HIRES2L  =$D5        HIRESH   =$D4        HIRESL   =$D3        HMOVE    =$6210
?  HPRINT   =$604C      HSCORE0  =$6007      HSCORE1  =$6008      HSCORE2  =$6009
   HSCORE3  =$600A      KEY      =$C000      LOOP1    =$6384      LOOP2    =$638F
   MANVERT  =$60DB   ?  MOVE1    =$621A      MOVE2    =$6230      MOVE2A   =$6249
   MOVE2B   =$6258      MOVE3    =$6267      MOVE3A   =$6280      MOVE3B   =$628F
   MOVE3C   =$629B      MOVE3D   =$62A4   ?  MOVE4    =$62B1      MOVE5    =$62C0
   MOVE6    =$62E3      MOVE7    =$6303      MOVEA    =$6224      MOVEB    =$62BF
   MOVEFLAG=$D0         MVFLAG   =$FB        NOCLICK  =$6118      NODELTA  =$6139
   ONE      =$FD        OUT      =$65AE      OUT1     =$65B1   ?  OVER     =$61BE
?  PAGE     =$614D      PAGE1    =$614F      PAGE2    =$6157      PASS     =$610F
   PAUSE    =$6599      PITCH    =$FF        PRINT    =$600B      RAINNO   =$E9
   RARRY    =$6143      RCOL1A   =$6479      RCOL1B   =$6480      RCOL1C   =$6489
   RCOL1D   =$6490      RCOL1E   =$6499      RCOL1F   =$64B4      RCOL2A   =$64D3
   RCOL2B   =$64DA      RCOL2C   =$64E3      RCOL2D   =$64EA      RCOL2E   =$64F3
   RCOL2F   =$650E      RCOL3A   =$652D      RCOL3B   =$6534      RCOL3C   =$653D
   RCOL3D   =$6544      RCOL3E   =$654D      RCOL3F   =$6568      RCOLLIDE1=$6475
   RCOLLIDE2=$64CF      RCOLLIDE3=$6529      READ     =$65A5      READPDL  =$61DD
   RESETKEY=$C010       RETURN   =$6392      RLOOP    =$61E8      RPATH1   =$945C
   RPITCH   =$D2     ?  RSHNUM   =$9409      RVERT    =$94BD   ?  RXBIT    =$94BC
   RXBYTE   =$94BB      RXTEMP   =$E8        S1       =$60DF      S2       =$60E6
   SCORE    =$608F      SCORE0   =$6003      SCORE1   =$6004      SCORE2   =$6005
   SCORE3   =$6006      SEND     =$60EE      SHFTREGH=$E1         SHFTREGL=$E0
   SHNUM    =$03        SOUND    =$60DC      SOUND0   =$629D      SPEAKER  =$C030
   START    =$634D   ?  STEP     =$65A2   ?  SWEEP    =$6120      TIME     =$FE
   TIMEH    =$E3        TIMEL    =$E2     ?  UPIT     =$612B      VERT     =$02
   VMOVE    =$62A7      W1       =$60F2      W2       =$60F9      WAIT     =$60EF
   WEND     =$6101      XBIT     =$01        XBYTE    =$00        XDRAW    =$63EB
   XDRAWMAN=$6324       XLOOPA   =$6414      XLOOPB   =$6451      XLOOPC   =$645C
   XLOOPD   =$6441      XLOOPE   =$644D      XLOOPF   =$6463      XLOOPG   =$6432
   YVERT2H  =$6700      YVERT2L  =$67C0      YVERTH   =$9200      YVERTL   =$92C0
   ZERO     =$FC
```

步步為營——TRIPPPLES

陳文緯

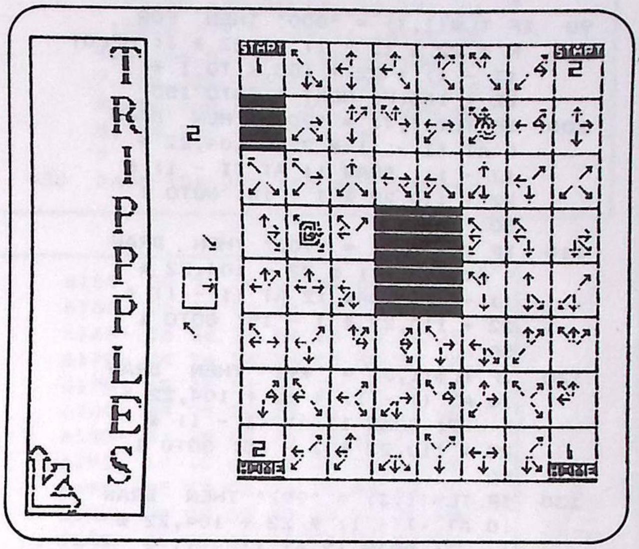
TRIPPLES 是一副很易學會怎樣玩的棋，但它事實上是很複雜的。這遊戲的目的是要由起點行至對角的終點（HOME）。

開始時，電腦會發出一些音樂，同時慢慢地畫出TRIPPPLES字樣。片刻後，螢幕右邊會繪出一個八行，每行八格，共六十四格的棋盤。遊戲可供兩人玩。

玩者１的棋子以正方形代表；玩者２的棋子以圓形代表。筆者建議有經驗的棋手用玩者１，初學的玩者用玩者２；你會慢慢發覺原因的。玩者１先行。

每格上有三個指出三個不同方向的箭咀。玩者行的時候，會跟着對手所處於的方格裏指着的方向而行。但是，棋盤的邊緣、空格、起點、對方的終點和對方的棋子會阻止你的棋子前進。所以，若你要勝出，你一定要設法迫你的對手去某格來使你能去某個較有利的格。

在選擇行那個方向，電腦會在棋盤旁顯示出可通行的方向，供玩者選擇。玩者可以用←或→鍵來移動方格，在適當的地方按 RETURN 鍵來結束選擇。如果棋子不能在三個方向中行任何一個，電腦



會發出聲響以作表示。

本程式共有兩個檔案：TRIPPLES 主程式及 TRIPPPLES SHAPE 圖形表。

大家在打入圖形表的時候可用下列指令存入磁碟：BSAVE TRIPPPLES SHAPE,A$6000，L$2C8。

有關機械碼程式的打入方法可參閱〔如何輸入機械碼〕一文。■

```
1   REM  ******************
2   REM  *    TRIPPPLES    *
3   REM  *   BY MANVILLE   *
4   REM  ******************
10  FOR I = 1 TO  PEEK (78) / 5:A
     =  RND (1): NEXT : IF  PEEK
    (24576) <  > 22 THEN  PRINT
     CHR$ (4);"BLOAD TRIPPPLES S
    HAPES"
20  DIM TL$(8,8),ST$(56),AR$(8),P
    (2,2): HGR : POKE  - 16302,0
    : POKE 232,0: POKE 233,96: SCALE=
    1: ROT= 0: HCOLOR= 3: FOR I =
    770 TO 790: READ J: POKE I,J
    : NEXT : HPLOT 0,0 TO 0,191 TO
    57,191 TO 57,0 TO 0,0: FOR I
     = 1 TO 41 STEP 5: FOR J = I
     TO I + 4
30  DRAW (16 + (I - 1) / 5) * (J <
    16) + (14 + (I - 1) / 5) * (
    J > 30) + 19 * (J > 15 AND J
     < 31) AT 40,4 * I: READ A,B
    : IF A = 0 THEN  FOR R = 1 TO
     B * 32: NEXT : GOTO 50
40  POKE 768,A: POKE 769,B * 32: CALL
     770
50  NEXT : NEXT : HPLOT 3,188: FOR
    I = 1 TO 20: READ A,B: HPLOT
     TO A,B: NEXT : FOR I = 1 TO
    56: READ ST$(I): NEXT : FOR
    I = 1 TO 8: READ AR$(I): NEXT
    : FOR I = 1 TO 56:R% =  RND
    (1) * 56 + 1:ST$ = ST$(I):ST
    $(I) = ST$(R%):ST$(R%) = ST$
    : NEXT :K = K + 1: FOR J = 1
     TO 8: FOR I = 1 TO 8
60  ON ((I = 1 OR I = 8) AND (J =
    1 OR J = 8)) OR ((I = 4 OR I
     = 5) AND (J = 4 OR J = 5)) GOTO
     70:TL$(I,J) = ST$(K):K = K +
    1
70  NEXT : NEXT :TL$(1,1) = "901"
    :TL$(8,1) = "902":TL$(1,8) =
    "992":TL$(8,8) = "991":TL$(4
    ,4) = "000":TL$(5,4) = "000"
    :TL$(4,5) = "000":TL$(5,5) =
    "000":P(1,1) = 1:P(1,2) = 1:
    P(2,1) = 8:P(2,2) = 1
```

```
80  FOR M = 1 TO 9: HPLOT 103,(M -
    1) * 22 TO 279,(M - 1) * 22:
     HPLOT (M - 1) * 22 + 103,1 TO
    (M - 1) * 22 + 103,175: NEXT
    : FOR J = 1 TO 8: FOR I = 1 TO
    8
90  IF TL$(I,J) = "000" THEN  FOR
    K = 22 * (J - 1) TO 22 * J: HPLOT
    (I - 1) * 22 + 103,K TO I *
    22 + 103,K: NEXT : GOTO 150
100 IF TL$(I,J) = "901" THEN  DRAW
    9 AT (I - 1) * 22 + 104,22 *
    (J - 1): DRAW 11 AT (I - 1) *
    22 + 110,22 * J - 15: GOTO 1
    50
110 IF TL$(I,J) = "902" THEN  DRAW
    9 AT (I - 1) * 22 + 104,22 *
    (J - 1): DRAW 12 AT (I - 1) *
    22 + 110,22 * J - 15: GOTO 1
    50
120 IF TL$(I,J) = "991" THEN  DRAW
    10 AT (I - 1) * 22 + 104,22 *
    J - 7: DRAW 11 AT (I - 1) *
    22 + 110,22 * J - 15: GOTO 1
    50
130 IF TL$(I,J) = "992" THEN  DRAW
    10 AT (I - 1) * 22 + 104,22 *
    J - 7: DRAW 12 AT (I - 1) *
    22 + 110,22 * J - 15: GOTO 1
    50
140 FOR K = 1 TO 3: DRAW  VAL ( MID$
    (TL$(I,J),K,1)) AT (I - 1) *
    22 + 114,22 * J - 11: NEXT
150 NEXT : NEXT : XDRAW 14 AT (P
    (1,1) - 1) * 22 + 110,P(1,2)
     * 22 - 15: XDRAW 15 AT (P(2
    ,1) - 1) * 22 + 110,P(2,2) *
    22 - 15
160 FOR P = 1 TO 2: FOR K = 1 TO
    9: XDRAW P + 10 AT 77,35: NEXT
    :OP$ = TL$(P(3 - P,1),P(3 -
    P,2)): IF OP$ = "902" THEN O
    P$ = "123"
170 IF P = 2 AND TL$(P(P,1),P(P,
    2)) = "902" THEN OP$ = "178"
180 TT$ = "": FOR K = 1 TO 3:F =
    0:S = P(P,1) +  ASC ( MID$ (
    AR$( VAL ( MID$ (OP$,K,1))),
    1,1)) - 66: IF S < 1 OR S >
    8 THEN 210
190 T = P(P,2) +  ASC ( MID$ (AR$
    ( VAL ( MID$ (OP$,K,1))),2,1
    )) - 66: IF T < 1 OR T > 8 THEN
    210
200 ON TL$(S,T) = "000" OR (TL$(
    S,T) <  > "99" +  STR$ (P) AND
    VAL (TL$(S,T)) > 900) OR (P
    (3 - P,1) = S AND P(3 - P,2)
    = T) GOTO 210:F =  VAL ( MID$
    (OP$,K,1))
210 TT$ = TT$ +  STR$ (F): NEXT :
     IF TT$ = "000" THEN  POKE 7
    68,192: POKE 769,180: CALL 7
    70: GOTO 320
220 FOR K = 1 TO 3: ON  MID$ (TT
    $,K,1) = "0" GOTO 230: DRAW
     VAL ( MID$ (TT$,K,1)) AT 82
    ,(K + 3) * 22 - 11
230 NEXT :L = 1: IF  MID$ (TT$,1
    ,1) = "0" THEN L = 2: IF  MID$
    (TT$,2,1) = "0" THEN L = 3
240 XDRAW 13 AT 74,(L + 3) * 22 -
    18
250 POKE  - 16368,0
260 K =  PEEK ( - 16384) - 128: ON
    K < 0 GOTO 260: ON K = 13 GOTO
    290: ON K <  > 21 AND K <  >
    8 GOTO 250:LL =  - 1: IF K =
    21 THEN LL = 1
270 ON L + LL < 1 OR L + LL > 3 GOTO
    250: IF  MID$ (TT$,L + LL,1)
     = "0" THEN LL = LL * 2: GOTO
    270
280 XDRAW 13 AT 74,(L + 3) * 22 -
    18:L = L + LL: GOTO 240
290 HCOLOR= 0: FOR I = 73 TO 92:
     HPLOT I,60 TO I,130: NEXT :
     HCOLOR= 3: XDRAW P + 13 AT
    (P(P,1) - 1) * 22 + 110,P(P,
    2) * 22 - 15
300 P(P,1) = P(P,1) +  ASC ( MID$
    (AR$( VAL ( MID$ (TT$,L,1)))
    ,1,1)) - 66:P(P,2) = P(P,2) +
     ASC ( MID$ (AR$( VAL ( MID$
    (TT$,L,1))),2,1)) - 66: XDRAW
    P + 13 AT (P(P,1) - 1) * 22 +
    110,P(P,2) * 22 - 15
310 ON TL$(P(P,1),P(P,2)) = "99"
     +  STR$ (P) GOTO 330
320 XDRAW P + 10 AT 77,35: NEXT
    : GOTO 160
330 CALL  - 1998: COLOR= 15: ON
    P = 2 GOTO 340: HLIN 15,22 AT
    15: HLIN 17,20 AT 17: HLIN 1
    9,20 AT 20: HLIN 17,22 AT 22
    : HLIN 15,24 AT 24: VLIN 15,
    24 AT 15: VLIN 17,22 AT 17: VLIN
    17,20 AT 20: VLIN 15,22 AT 2
    2: VLIN 15,24 AT 24: GOTO 36
    0
340 PLOT 24,23: HLIN 19,23 AT 24
    : HLIN 17,18 AT 23: VLIN 21,
    22 AT 16: VLIN 19,20 AT 15: VLIN
    17,18 AT 16: HLIN 17,18 AT 1
    6: HLIN 19,20 AT 15
350 HLIN 21,22 AT 16: VLIN 17,18
     AT 23: PLOT 24,19: VLIN 20,
    21 AT 23: HLIN 20,22 AT 22: PLOT
    19,21: VLIN 19,20 AT 18: HLIN
    19,20 AT 18: VLIN 19,20 AT 2
    1
360 POKE  - 16368,0
370 POKE  - 16298,0: ON  PEEK ( -
    16384) > 127 GOTO 380: POKE
     - 16297,0: GOTO 370
380 TEXT : HOME : VTAB 10: PRINT
    "DO YOU WANT TO PLAY AGAIN ?
     ( Y OR N ) ";
390 GET A$: IF A$ = "Y" THEN  RUN
```

```
400  ON A$ <  > "N" GOTO 390: TEXT
     : HOME : END : DATA 173,48,1
     92,136,208,5,206,1,3,240,9,2
     02,208,245,174,0,3,76,2,3,96
     ,192,2,145,2,115,4,115,2,115
     ,2,115,4,128,2,115,2,128,2,1
     45,0,0,1,0,1
410  DATA 192,2,145,2,115,4,145,2
     ,115,2,96,4,108,2,115,2,128,
     0,0,1,0,1,96,2,108,2,115,4,1
     15,2,128,2,145,4,128,2,115,2
     ,96,2,108,0,0,1,0,1,152,2,17
     2,2,192,4,152,2,145,2,128,4,
     115,2,128,2,145,0,0,2
420  DATA 21,188,21,191,26,186,26
     ,185,21,180,21,183,12,183,18
     ,177,20,179,20,171,12,171,14
     ,173,8,179,8,169,11,169,6,16
     4,5,164,0,169,3,169,3,188,14
     7,126,256,238,147,235,157,15
     8,356,124,368,357,123,278,24
     5,135,145,456,246,128,348,25
     7
430  DATA 156,247,134,467,345,457
```

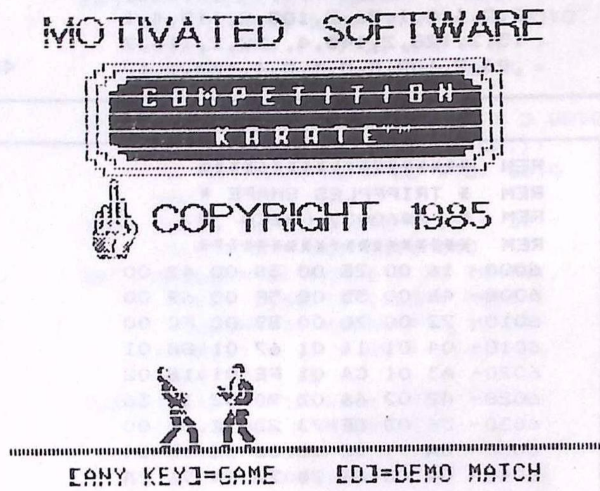
```
REM  *********************
REM  * TRIPPPLES SHAPE *
REM  *  A$6000, L$2C8  *
REM  *********************
6000- 16 00 2E 00 38 00 42 00
6008- 4B 00 55 00 5F 00 69 00
6010- 72 00 7C 00 B9 00 FC 00
6018- 04 01 11 01 67 01 88 01
6020- A3 01 CA 01 FE 01 1B 02
6028- 42 02 66 02 9D 02 12 36
6030- 36 0D DF 73 2D 1E 04 00
6038- 8A 71 0E 0E 0E 3E 0D 24
6040- 04 00 09 2D 2D E4 96 2A
6048- 28 20 00 41 0C 0C 0C 0C
6050- 3C 0D 36 04 00 C1 20 24
6058- 2C 15 DF 63 0C 04 00 C3
6060- E0 1C 1C 1C 27 2D D6 23
6068- 00 1B 3F 3F 64 96 3A E0
6070- 04 00 9A F3 1E 1E 1E 27
6078- 16 2D 04 00 2D 2D 2D 2D
6080- 2D 2D 2D 2D 2D 2D DE FB
6088- DB FF 3B DF 33 3F 4D 0D
6090- 6D 6D 0D 6D 6D FE FF DB
6098- DF 1B FF F7 1B 6D 6D 6D
60A0- 6D 0D 0D 2D 0D FE FF FF
60A8- 1F FF FF FF 13 2D 2D 2D
60B0- 2D 2D 2D 2D 2D 2D 2D 04
60B8- 00 2D 2D 2D 2D 2D 2D 2D
60C0- 2D 2D 2D DE 1B 1F 3F 1F
60C8- FF 1B FF 37 6D 0D 2D 0D
60D0- 4D 4D 0D 2D FE 1B 1F 1F
60D8- 1F 1F 3F 1F DF 2A 0D 0D
60E0- 2D 0D 0D 0D 0D 0D 2D DE
60E8- 1B 1F 3F 1F FF 1B FF BF
60F0- 2D 2D 2D 2D 2D 2D 2D 2D
60F8- 2D 2D 04 00 09 36 36 36
6100- 25 24 04 00 2D 2D 35 36
6108- 3F 3F 37 35 37 2D 2D 25
6110- 00 2D 2D 2D 2D 2D 2D 2D
6118- 2D 36 36 36 36 36 36 36
6120- 36 3F 3F 3F 3F 3F 3F 3F
6128- 3F 24 24 24 24 24 24 24
6130- 24 00 00 00 00 00 00 00
6138- 00 00 00 00 00 00 00 00
6140- 00 00 00 00 00 00 00 00
6148- 00 00 00 00 00 00 00 00
6150- 00 00 00 00 00 00 00 00
6158- 00 00 00 00 00 00 00 00
6160- 00 00 00 00 00 00 00 36
6168- 36 36 36 2E 2D 2D 2D 2D
6170- 24 24 24 24 FC 3F 3F 3F
6178- 4E 49 31 36 36 3E 3F 3F
6180- 24 24 2C 2D 36 3E 04 00
6188- 49 29 0E 75 76 1E F6 3F
6190- 1C 1C 64 AD 3E 18 18 18
6198- 08 F7 F6 76 76 AD 2D 2D
61A0- 0C 04 00 32 66 2C 3C 0D
61A8- 2D 35 36 36 36 36 36 36
61B0- 3E 0D 2D 1C 27 25 27 25
61B8- 27 25 27 25 27 25 27 25
61C0- 27 2D 2D 2D 3E 0D 3E 15
61C8- 04 00 35 36 36 36 36 36
61D0- 36 3E 0D 2D 1C 27 25 27
61D8- 25 27 25 27 25 27 25 27
61E0- 25 27 2D 2D 2D 2E 2E 3E
61E8- 0D 3E 37 F5 37 3F 3F 4D
61F0- 32 25 35 3E 0D 3E 2E 25
61F8- 36 6F 3E 0D 25 00 49 09
6200- 2D 2D 1E 36 36 36 36 36
6208- 36 2E 1F 24 24 24 24 24
6210- 24 3C 36 36 36 36 36 36
6218- 3E 04 00 35 36 36 36 36
6220- 36 36 3E 0D 2D 1C 27 25
6228- 27 25 27 25 27 25 27 25
6230- 27 25 27 2D 2D 2D 2E 2E
6238- 2E 1F 2E F5 37 37 3F 3F
6240- 04 00 2D 2D 1E 36 36 36
6248- 36 36 36 3E 24 24 24 24
6250- 24 24 3C 36 36 36 36 36
6258- 36 3E 4D 29 2D 2D 2D 2D
6260- 2C 24 1E 1E 04 00 2D 2D
6268- 2D 2D 2D 2D 35 35 E6 1C
6270- DF DB DB 1B 36 36 36 36
6278- 36 36 3E 0D 24 24 24 24
6280- 24 24 2C 36 36 36 2D 2D
6288- 25 25 16 36 1C DF 1B 36
6290- 36 36 2D 2D 2D 2D 2D 2C
6298- 24 1E 1E 04 00 49 2D 2D
62A0- 35 35 35 24 FC DB DB 32
62A8- 37 35 37 2D 34 2E FD 33
62B0- 2D 2D 2D 3E 3F 3F 4E 2D
62B8- 2D 3E 0D 3E 2E 1E F7 3F
62C0- 3F 1C 3F 26 24 2E 04 00
62C8- FF
```

1. HERO
2. SPIDER MAN
3. WINNIE THE POOH
4. THE ISLAND OF BETA
5. COMPETITION KARATE



最新遊戲軟件評價

李德華



遊戲名稱：COMPETITION KARATE

遊戲類型：ACTION

評　　分：三粒星半

繼 KARATEKA 後，坊間又再推出另一空手道的搏擊遊戲——COMPETITION KARATE。

在遊戲開始前，玩者可選擇DEMO，這可令玩者更容易明白遊戲的玩法，若果不選擇 DEMO，便會出現遊戲的 MENU。

（1）CREATE NEW STUDENT

如果玩者是第一次玩的話，在玩之前必定要選這項功能，首先要輸入姓名，再而用遊戲桿選擇玩者的反應、強壯、高度、重量、攻擊力等等。選擇完畢後，便返回MENU。

（2）WORK OUT IN THE DOJO

DOJO 是一間空手道的學校，玩者可在比賽後到這裡把認爲要改進的地方改進過來，而改進的方法是同第一項選擇一樣。

（3）STAGE A TOURNAMENT

選擇這項功能，便準備和對手決戰，決戰的方法有兩種，一種是除制賽，而另一種是沒有計分的比賽，通常選擇○，這時會出現另一MENU，分別是：

〔1〕WHITE
〔2〕BLUE
〔3〕BROWN
〔4〕BLACK

選擇1（如果你在除制賽勝了，便會升級，這時便可選擇2、3或4），當電腦把需要的資料讀入後，便打出放入DATA DISK的訊息，當一切準備就緒後，電腦會要求玩者打入對手的名字，而玩者最少要輸入四個人的名字，電腦跟着又會打出放入遊戲碟的訊息，大約等十五秒鐘，電腦又會詢問玩者使用鍵盤或遊戲桿進行遊戲，由於問題太多，筆者再不詳細說明，相信聰明的讀者也懂得回答了，再而便是遊戲正式開始，我方和敵方會出現在比武台上，而台下的觀衆便是你們的師兄、師父及師弟，他們正鼓勵你把對方擊倒，現在就用你師父所教的六招絕招和對方比拼吧！

KEYBOARD 控制分佈圖

左面	招式	右面
1 2 3	1 2 3	8 9 0
Q W E	4 5 6	I O P
A S D	7 8 9	K L ;
ESC		RETURN

招式：

1—PUNCH
2—BACKFIST
3—LUNGE
4—WATCH ARM
5—RETREAT（撤退）
6—WATCH LEGS
7—FRONT KICK
8—SIDE KICK
9—SPIN KICK

玩者可用己方的KEYBOARD來使出不同的招數，例如你是左方，那麼請先把箭咀的數字移去（1、2、3、7、8、9），然後再按ESC便能出招攻擊對方。

當兩隊隊員比賽後，便會產生兩隊的冠軍，而這兩隊的冠軍會再決鬥，最後便能夠得到大賽的冠軍。

除了隊制賽外，還有不設計分制的比賽，這比賽可以令玩者作賽前練習。

（4）REVIEW TOP IO FIGHTERS

當玩者比賽後，你的紀錄便會寫入碟中，而玩者可以利用這項功能進行觀察。

（5）USE PROGRAM UTILITIES（原裝貨才有）

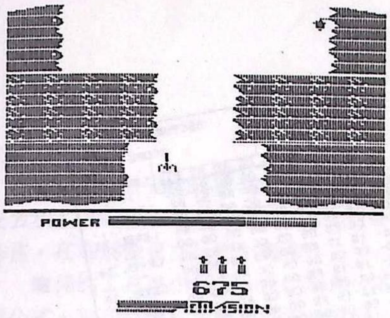
這功能可編輯及印出每隊的名單、觀察及印出學生的統計紀錄……等等，但如果你的磁碟不是原裝貨的話，便不能作以上的工作。

（6）SAUPMARRA

離開遊戲

後話：經筆者多番調查下，發覺在坊間流傳的版本多不勝數，大多數是沒有第五項功能及不能在遊戲中CATALOG，有些版本甚至不能夠拿到奬杯，無論怎樣也好，COMPETITION KARATE 確是值得推薦的遊戲。■





遊戲名稱：中華英雄（HERO）
遊戲類型：ACTION
評　　分：二粒半星

HERO 是ADVISION繼捉鬼敢死隊後的另一同廠出品。

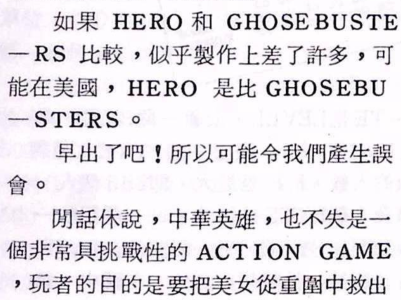
如果 HERO 和 GHOSEBUSTE—RS 比較，似乎製作上差了許多，可能在美國，HERO 是比 GHOSEBU—STERS。

早出了吧！所以可能令我們產生誤會。

閒話休說，中華英雄，也不失是一個非常具挑戰性的 ACTION GAME，玩者的目的是要把美女從重圍中救出，但途中隱藏了不少殺機，繼有不同的怪物襲擊、電光牆、狹窄的山路、高難度的狹窄山路及流沙等等。

雖然 HERO 有數不盡的難關，但玩者配備了永遠發不盡的激光槍，而且還有隨身配備的小型炸彈，縱觀以上的裝備，想救出美女根本不成問題。

這遊戲只有一個難處就是有很多〝隱形”的陷阱，玩者能否順利保護出美女，就要靠你自己了。■

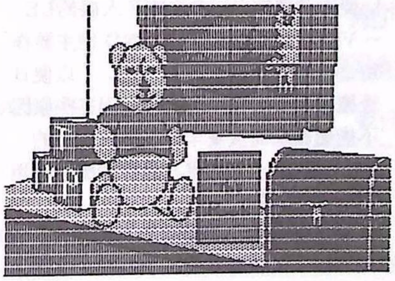


遊戲名稱：THE ISLAND OF BETA
遊戲類型：ADVENTURE
評　　分：二粒星

玩過 DALLAS QUEST 的讀者，都認爲尋寶的 ADVENTURE GA—ME 非常具挑戰性，坊間最近又推出另一尋寶的 ADVENTURE—THE ISLAN OF BETA。

遊戲的目的當然是尋找實藏，而在島上分佈了不同的實箱，它們分別劃分起 BETA ISLAND的島嶼上。

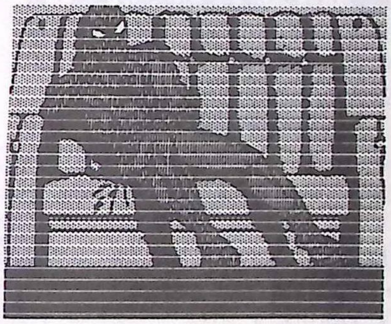
玩者可在遊戲中得知島嶼的分佈地圖，所有道路一經經過，便會毀壞不堪及永遠不能再通過，所以玩時應細心考慮才輸入要行的方向。鑑於近期ADV—ENTURE GAME 頻頻出現，筆者未能專心分析破解，留待日後有機會再談吧！■



遊戲名稱：WINNIE THE POUH
遊戲型式：ADVENTURE
評　　分：二粒半星

有一天，森林上忽然刮起大風，把所有動物的東西也吹走了，經過動物們細心挑選後，便認爲小熊熊是最佳尋找牠們失物的人選，現在就用你的聰明智慧把失物尋找回來吧！

小熊熊歷險記是非常容易玩的 AD—VENTURE GAME，輸入答案的方法就如 DARK CRYSTAL〕〔一樣，所不同的是，玩者還要選擇東、南、西、北的路去歷險，玩者遇到困難的時候，可到貓頭鷹那處去詢問遊戲的提示，這遊戲最困難的地方就是每一次玩，都會把東西重新分件及編排，所以並不是每次玩法一樣，這一點讀者要注意。當玩者把十樣東西送歸原主後，便應到 TABLE處，這時動物們便會爲你開慶祝大會，遊戲亦隨之完結。經筆者一番破解後 APPLE FAN CLUB 會員可在每月通訊上留意。■



遊戲名稱：SPIDER MAN
遊戲類型：ACTION ADVENTURE
評　　分：三粒星

看過蜘蛛俠卡通的讀者，一定非常羡慕蜘蛛俠擁有爬牆、放蜘蛛絲等本能最重要是把壞人或怪獸消滅。

蜘蛛俠 ADVENTURE GAME 是 HULK 變形俠醫的同廠出品，所以很多玩法也極爲相似。

遊戲開始時，你已經是一名蜘蛛俠，任務是找回失落的 GEM，但是很多怪獸會阻止你這項任務，要消滅這些怪獸，並不是單靠蜘蛛絲，還要加上自己的聰明和智慧，筆者經幾番努力，已拿了不粒寳石及消滅了兩隻怪獸，在這裏筆者不妨給些提示給玩者。提示如下：

大多數寳石都在電梯槽內，有些在怪獸身上、抽屜裏等等，消滅第一隻怪獸是要把化學物體混合起來，然後把液體倒在怪獸身上便消滅了，第二隻怪獸是要把溫度降至最低，怪獸便會僵硬起來，那麽就可消滅了第二隻怪獸。■

玩CHAMPION SHIP
LODE RUNNER 150 LEVELS

陳百儀

LODE RUNNER 是一個很考智慧的遊戲，自從LODE RUNNER 推出之後，筆者便極力向朋友推薦，是一個設計得非常優良的遊戲，例如它可以清洗最高積分，自選LEVEL ，選擇任何的LEVEL 玩等等，而到CHAMPIONS—HIP LODE RUNNER 面世後，更令筆者鼓舞，因爲它的難度更高，更考智慧，但不知讀者會否因難度過高，祇是時常玩LEVEL 1 ，而放棄這個遊戲呢？如是就眞可惜，本文是介紹各位蘋果遊戲迷如何可以玩到其他LEVEL。

要直接玩LEVEL 2或以上，是須要更改磁碟某一條軌的內容，這是不困難的，但在未講怎樣更改磁碟內容之前，本人想講述在CHAMPIONSHIP LODE RUNNER之其中二個COMM—AND ，一是 CTRL—S，另一是CTRL—L 。

CTRL—S 一般是用作控制聲音，但在那個MAN 在閃動的時候（例如當在新LEVEL 剛剛開始時，或剛剛死了一個MAN 之後），按CTRL—S ，跟着按S ，便可SAVE 你在玩着的LEVE—L 入碟，日後開機玩時，按CTRL—L 便可選擇你以前曾SAVE 入碟的LE—VEL，這兩個COMMAND 的主要作用是暫時儲起玩着的LEVEL ，以便日後繼續玩，於是，玩者不會因某些原因不能繼續玩而放棄辛辛苦苦破解了的LEVEL 了。讀者現在自行開機熟習用這兩個COMMAND。

```
TRACK = $0C                SECTOR = $0D

00- CC C5 D6 C5 CC A0 A0 B2  LEVEL  2
08- 02 01 FF 00 00 00 00 00  .. .....
10- CC C5 D6 C5 CC A0 A0 B3  LEVEL  3
18- 03 02 FF 00 00 00 00 00  .. .....
20- CC C5 D6 C5 CC A0 A0 B4  LEVEL  4
28- 04 03 FF 00 00 00 00 00  .. .....
30- CC C5 D6 C5 CC A0 A0 B5  LEVEL  5
38- 05 04 FF 00 00 00 00 00  .. .....
40- CC C5 D6 C5 CC A0 A0 B6  LEVEL  6
48- 06 05 FF 00 00 00 00 00  .. .....
50- CC C5 D6 C5 CC A0 A0 B7  LEVEL  7
58- 07 06 FF 00 00 00 00 00  .. .....
60- CC C5 D6 C5 CC A0 A0 B8  LEVEL  8
68- 08 07 FF 00 00 00 00 00  .. .....
70- CC C5 D6 C5 CC A0 A0 B9  LEVEL  9
78- 09 08 FF 00 00 00 00 00  .. .....
80- CC C5 D6 C5 CC A0 B1 B0  LEVEL 10
88- 0A 09 FF 00 00 00 00 00  .. .....
90- CC C5 D6 C5 CC B1 B5 B0  LEVEL150
98- 96 95 FF 00 00 00 00 00  .. .....
A0- 00 00 00 00 00 00 00 00  ........
A8- 00 00 00 00 00 00 00 00  ........
B0- 00 00 00 00 00 00 00 00  ........
B8- 00 00 00 00 00 00 00 00  ........
C0- 00 00 00 00 00 00 00 00  ........
C8- 00 00 00 00 00 00 00 00  ........
D0- 00 00 00 00 00 00 00 00  ........
D8- 00 00 00 00 00 00 00 00  ........
E0- 00 00 00 00 00 00 00 00  ........
E8- 00 00 00 00 00 00 00 00  ........
F0- 00 00 00 00 CC CF C4 C5  ....LODE
F8- A0 D2 D5 CE CE C5 52 00  RUNNE..
```

Fig. 1

當讀者明白這兩個COMMAND 的操作後，更改辦法現在便開始，其理論上是更改玩者已SAVE 入碟的LEVEL，例如本是玩到LEVEL 2 ，用CTRL—S 及S SAVE 入碟後，改變磁碟的內容，使它變成其他LEVEL 即可，譬如變成LEVEL 10或LEVEL 80等等。

要更改磁碟的內容，是要靠一些軟件的幫助，例如COPY ］〔 PLUS 4.0之類，筆者不在此詳述，但最主要的是改變TRACK ＄0C，SECTOR ＄0D的內容，要知怎樣改，便要先知道它記錄的格式。

在TRACK ＄0C，SECTOR ＄0D，可存十個記錄，每個記錄須16BYTE，首8 BYTE是名字，可以是COMPU—TER，PATRICK，ABC，甚至直接叫LEVEL 21，如圖二，跟着一個BY—TE是LEVEL，跟着一個BYTE 是LEVEL —1 ，跟着一個BYTE 是剩餘的人數，F F 是最大，即255個人，隨後 5 個BYTE是積分，每一個記錄一共須要16BYTE，下一個記錄在BYTE ＄10開始，見圖便更清楚。在圖中，首8 個BYTE是名字，可以是任何名稱，主要是更改BYTE ＄08 及BYTE ＄09，BYTE ＄18及BYTE 19 … 等等，因爲它們就是眞正的LEVEL 數值。如果照圖三中更改，便可玩LEVEL 31 至LEVEL 40其中任何一個。當讀者明白如何更改，便可以玩心目中任何一個的LEVEL ，但最高祇可改到LEVEL 150 。

```
TRACK = $0C                SECTOR = $0D

00- CC C5 D6 C5 CC A0 B3 B1  LEVEL 31
08- 1F 1E FA 00 00 00 00 00  ..z.....
10- CC C5 D6 C5 CC A0 B3 B2  LEVEL 32
18- 20 1F FD 00 00 00 00 00  ..}.....
20- CC C5 D6 C5 CC A0 B3 B3  LEVEL 33
28- 21 20 FE 00 00 00 00 00  ..~.....
30- CC C5 D6 C5 CC A0 B3 B4  LEVEL 34
38- 22 21 FE 00 00 00 00 00  ..~.....
40- CC C5 D6 C5 CC A0 B3 B5  LEVEL 35
48- 23 22 FE 00 00 00 00 00  ..~.....
50- CC C5 D6 C5 CC A0 B3 B6  LEVEL 36
58- 24 23 FE 00 00 00 00 00  ..~.....
60- CC C5 D6 C5 CC A0 B3 B7  LEVEL 37
68- 25 24 FE 00 00 00 00 00  ..~.....
70- CC C5 D6 C5 CC A0 B3 B8  LEVEL 38
78- 26 25 FD 00 00 00 00 00  ..}.....
80- CC C5 D6 C5 CC A0 B3 B9  LEVEL 39
88- 27 26 FE 00 00 00 00 00  ..~.....
90- CC C5 D6 C5 CC A0 B4 B0  LEVEL 40
98- 28 27 FD 00 00 00 00 00  ..}.....
A0- 00 00 00 00 00 00 00 00  ........
A8- 00 00 00 00 00 00 00 00  ........
B0- 00 00 00 00 00 00 00 00  ........
B8- 00 00 00 00 00 00 00 00  ........
C0- 00 00 00 00 00 00 00 00  ........
C8- 00 00 00 00 00 00 00 00  ........
D0- 00 00 00 00 00 00 00 00  ........
D8- 00 00 00 00 00 00 00 00  ........
E0- 00 00 00 00 00 00 00 00  ........
E8- 00 00 00 00 00 00 00 00  ........
F0- 00 00 00 00 CC CF C4 C5  ....LODE
F8- A0 D2 D5 CE CE C5 52 00  RUNNE..
```

Fig. 3

```
TRACK = $0C                SECTOR = $0D
                 NAME              SCORE
          LEVEL
          LEVEL - 1
          MEN
00- CC C5 D6 C5 CC A0 B2 B1  LEVEL 21
08- 15 14 FD 00 00 00 00 00  ..}.....
10- CC C5 D6 C5 CC A0 B2 B2  LEVEL 22
18- 16 15 FE 00 00 00 00 00  ..~.....
20- CC C5 D6 C5 CC A0 B2 B3  LEVEL 23
28- 17 16 FE 00 00 00 00 00  ..~.....
30- CC C5 D6 C5 CC A0 B2 B4  LEVEL 24
38- 18 17 FE 00 00 00 00 00  ..~.....
40- CC C5 D6 C5 CC A0 B2 B5  LEVEL 25
48- 19 18 FE 00 00 00 00 00  ..~.....
50- CC C5 D6 C5 CC A0 B2 B6  LEVEL 26
58- 1A 19 FE 00 00 00 00 00  ..~.....
60- CC C5 D6 C5 CC A0 B2 B7  LEVEL 27
68- 1B 1A FE 00 00 00 00 00  ..~.....
70- CC C5 D6 C5 CC A0 B2 B8  LEVEL 28
78- 1C 1B FE 00 00 00 00 00  ..~.....
80- CC C5 D6 C5 CC A0 B2 B9  LEVEL 29
88- 1D 1C FE 00 00 00 00 00  ..~.....
90- CC C5 D6 C5 CC A0 B3 B0  LEVEL 30
98- 1E 1D FE 00 00 00 00 00  ..~.....
A0- 00 00 00 00 00 00 00 00  ........
A8- 00 00 00 00 00 00 00 00  ........
B0- 00 00 00 00 00 00 00 00  ........
B8- 00 00 00 00 00 00 00 00  ........
C0- 00 00 00 00 00 00 00 00  ........
C8- 00 00 00 00 00 00 00 00  ........
D0- 00 00 00 00 00 00 00 00  ........
D8- 00 00 00 00 00 00 00 00  ........
E0- 00 00 00 00 00 00 00 00  ........
E8- 00 00 00 00 00 00 00 00  ........
F0- 00 00 00 00 CC CF C4 C5  ....LODE
F8- A0 D2 D5 CE CE C5 52 00  RUNNE..
```

Fig. 2

這個遊戲的每個LEVEL 都有其獨特破解之方法，有的很容易，有的很難，極之考智商，例如LEVEL 31便是，筆者還未破解得到。另外當玩完LEVEL 150 後（低些LEVEL 亦可），會有些恭賀的賀辭出現，但這有待讀者自己體驗了。

最後，祝各位讀者更改成功，與筆者分享玩高LEVEL 的樂趣，及如有人能知如何破解LEVEL 31或其他很艱深的LEVEL ，請公諸同好。■

運用BASIC在高解像畫面玩魔術

BISCUIT

看了第十五期的「將高解像畫面倒轉」及「程式技巧—查表法」，靈機一觸，也嘗試用BASIC——最易懂的電腦語言，在高解像畫面玩些小魔術。

魔術的工具是由程式(一)及(二)所組成的表列。程式(一)是利用公式，將高解像畫面暫存區的垂直位址的關係算出，並製成垂直位址座標的表列。程式(二)產生一個「鏡面對稱」的表列，它將一個位元組內的位元由0至6依次序左右互換，位元7則保持不變，因此圖像的顏色維持不變。

程式(三)利用以上兩個程式所產生的表列，將任意大小的高解像畫面產生一個鏡面對稱的映像。例如左手會變成右手。首先，程式要求輸入圖像名稱，然後輸入畫面範圍。水平範圍以位元組爲單位，由0至39，共40組；垂直範圍以列爲單位，由0至191，共192列。程式將圖像輸入第二頁，再在第一頁輸出鏡面對稱的圖像。

行號500是此程式的靈魂，改變此行，便能變出很多把戲來。例如：

500 POKE V1 (191－I) ＋J，Z

可將任意大小的畫面倒轉；又如

500 POKE V1 (I) ＋J，255－Z

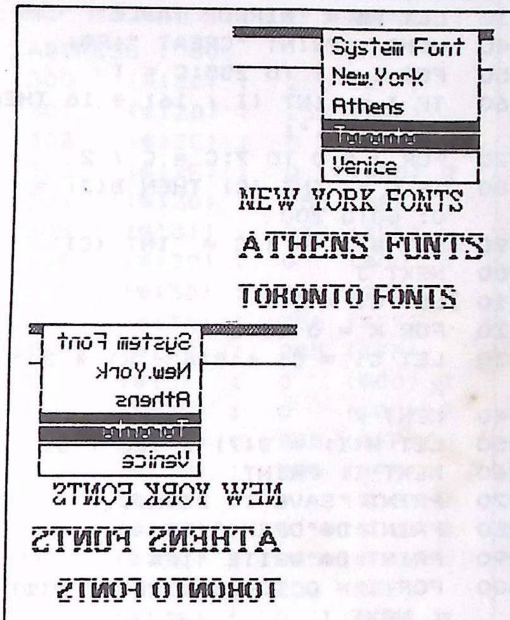
會使任意大小的畫面產生負映像，就像相機的負片一樣；又例如：

500 POKE V1 (I) ＋J，Z

可將第二頁的圖像作選擇性的拷貝往第一頁。

由於程式全用BASIC寫成，執行速度會慢一些，但程式可以輕易改寫及改良，各位可嘗試將程式作少許改動，使高解像畫面上下或左右繞捲，或者你比我做得更好呢。

本文共有三個程式，其檔案名字分別爲：



執行本程式後產生出來的圖像反面效果

程式(1)：CREAT VERTICAL TABLE
程式(2)：CREAT MIRROR TABLE
程式(3)：MIRROR PIC

在執行主程式(3)之前，必須要先RUN 程式(1)及程式(2)，使分別產生兩個TEXT FILE。■

```
10  REM  *********************
20  REM  CREAT VERTICLE TABLE
30  REM  *********************
110  CLEAR :A =  FRE (0):D$ =  CHR$
     (13) +  CHR$ (4)
120  DIM V(191)
130  LET F$ = "VERTICAL TABLE"
140  HOME : PRINT "CREAT ";F$;
150  FOR Y = 0 TO 191
160  IF Y =  INT (Y / 16) * 16 THEN
      PRINT ".";
170 Y1 =  INT (Y / 8):YR = Y - Y1
     * 8
180 Y2 =  INT (Y1 / 8):YS = Y1 -
     Y2 * 8
190 V(Y) = 8192 + Y2 * 40 + YS *
     128 + YR * 1024
200  NEXT Y: PRINT
210  PRINT "SAVE TO DISK..."
220  PRINT D$"OPEN ";F$
230  PRINT D$"WRITE ";F$
240  FOR I = 0 TO 191: PRINT V(I)
     : NEXT I
250  PRINT D$"CLOSE"
260  PRINT "JOB COMPLETED."
```

程式(1)

程式(2)

```
90  REM   ******************
100  REM  CREAT MIRROR TABLE
105  REM  ******************
110  CLEAR :A =  FRE (0):D$ =  CHR$
  (13) +  CHR$ (4)
120  DIM M(255)
130  LET F$ = "MIRROR TABLE"
140  HOME : PRINT "CREAT ";F$;
150  FOR I = 1 TO 255:C = I
160  IF I = INT (I / 16) * 16 THEN
  PRINT ".";
170  FOR J = 0 TO 7:C = C / 2
180  IF C =  INT (C) THEN B(J) =
 0: GOTO 200
190  LET B(J) = 1:C =  INT (C)
200  NEXT J
210  LET C1 = 0
220  FOR K = 0 TO 6
230  LET C1 = C1 + B(6 - K) * 2 ^
 K
240  NEXT K
250  LET M(I) = B(7) * 128 + C1
260  NEXT I: PRINT
270  PRINT "SAVE TO DISK..."
280  PRINT D$"OPEN ";F$
290  PRINT D$"WRITE ";F$
300  FOR I = 0 TO 255: PRINT M(I)
 : NEXT I
310  PRINT D$"CLOSE"
320  PRINT "JOB COMPLETED."
```

程式(3)

```
100  REM  PROGRAM: MIRROR PIC
110  REM   AUTHOR: BISCUIT
120  TEXT : HOME : POKE 216,0: CLEAR
 :A =  FRE (0)
130  DIM M(255),V1(191),V2(191)
140  LET D$ =  CHR$ (13) +  CHR$
 (4):F1$ = "MIRROR TABLE":F2$
  = "VERTICAL TABLE"
150  HTAB 13: PRINT "MIRROR PICTU
 RE"
160  VTAB 12: HTAB 10: FLASH : PRINT
 "  PROGRAM  LOADING  ": NORMAL
 : PRINT
170  PRINT D$"VERIFY";F1$
180  PRINT D$"VERIFY";F2$
190  PRINT D$"OPEN "F1$
200  PRINT D$"READ "F1$
210  FOR I = 0 TO 255: INPUT M(I)
 : NEXT
220  PRINT D$"CLOSE"
230  PRINT D$"OPEN "F2$
240  PRINT D$"READ "F2$
250  FOR I = 0 TO 191: INPUT V1(I
 ):V2(I) = V1(I) + 8192: NEXT

260  PRINT D$"CLOSE"
270  TEXT : HOME
280  VTAB 22: PRINT " ENTER THE N
 AME OF PICTURE (RETURN FOR
  CATALOG):";
290  PRINT "...................
 ......."
300  VTAB 23: HTAB 11: INPUT "";F
 $
310  IF F$ = "" THEN  HOME : PRINT
 D$;"CATALOG": GET A$: PRINT
 : PRINT : PRINT : GOTO 280
320  PRINT D$"VERIFY ";F$
330  INPUT "HORIZONTAL START (0-3
 9): ";X1
340  IF X1 < 0 OR X1 > 39 THEN  PRINT
  CHR$ (7): GOTO 330
350  PRINT
360  INPUT "HORIZONTAL END (0-39)
 : ";X2
370  IF X2 < 0 OR X2 > 39 THEN  PRINT
  CHR$ (7): GOTO 360
380  PRINT
390  INPUT "VERTICAL START (0-191
 ): ";Y1
400  IF Y1 < 0 OR Y1 > 191 THEN  PRINT
  CHR$ (7): GOTO 390
410  PRINT
420  INPUT "VERTICAL END (0-191):
  ";Y2
430  IF Y2 < 0 OR Y2 > 191 THEN  PRINT
  CHR$ (7): GOTO 420
440  HGR : HGR2 : POKE  - 16302,0

450  PRINT D$"BLOAD ";F$;",A$4000
 "
460  POKE  - 16300,0
470  FOR I = Y1 TO Y2
480  FOR J = X1 TO X2
490  LET Z =  PEEK (V2(I) + J)
500  POKE V1(I) + 39 - J,M(Z)
510  NEXT J,I
520  HOME : POKE  - 16301,0
530  VTAB 22: PRINT "GOODBYE!!!"
540  PRINT "TYPE 'GOTO 270' TO RE
 START"
```

簡單實用的

對照性機械碼列表

TIM

你有沒有試過打錯BINARY DATA 嗎？你有沒試過在打錯DATA 時，需要查這個DATA 的十進制、十六進制或ASCⅡ 數目，而覺得很麻煩和費時呢？

如果有上述的事情經常發生，機械碼列表（BINARY LISTER）對你的用處就不少了。因爲他可以將每一個AD—DRESS 裏的資料顯示在螢光幕，並顯示ADDRESS 裏的十進制和十六進制數字。ADDRESS 裏的資料裏顯示十進制、十六進制和ASCⅡ 碼，以方便用者對換。還可以連續顯示，直到你想停下爲止（圖一）。

當程式進行時，電腦會立刻清除畫面，並且顯示ADD—RESS：字，目的是要求用者輸入想找的ADDRESS ，可以按入十進制或十六進制；在按入十六進制數碼之前，必須鍵入＄字，以告訴電腦這個數字是否十六進制。RETURN被按入後，BINARY LISTER 便開始工作（顯示BINARY LIST 如圖一），每次列出二十個連續數目的BINARY AD—DRESS，並於螢光幕底下裏詢問用者是否希望將ADDR—ESS 繼續顯示，用者只須單按Y代表是，N代表否就可以照你所想去做了。

大家打入列表後，可用下列指令存入磁碟，或在今期出版的程式磁碟裏找尋相同名字的FILENAME：

SAVE BINARY LISTER

```
1   REM  ******************
2   REM  * BINARY LISTER *
3   REM  *    BY  TIM     *
4   REM  ******************
10 C$ = "0123456789ABCDEF": TEXT
     : HOME
20  VTAB 10: INPUT "ADDRESS : ";E
     $:E =  VAL (E$): IF  LEFT$ (
     E$,1) = "$" THEN A$ =  RIGHT$
     (E$, LEN (E$) - 1): GOSUB 11
     0:E = N
30  ON E < 0 OR E > 65536 GOTO 20
     : HOME :I = 0
40 C = I + E: GOSUB 80:B$ = A$:C =
      PEEK (I + E): GOSUB 80: PRINT
     E + I;: HTAB 7: PRINT "($"B$
     ") :";: HTAB 17: PRINT C;: HTAB
     21: PRINT "($"A$")";: HTAB 2
     7: IF C < 64 AND C =  > 0 THEN
      INVERSE :C = C + 64: GOTO 6
     0
50  IF C > 63 AND C < 128 THEN  FLASH
```

圖(1)　利用本文程式列印出來的機械碼列表例子。

```
ADDRESS : 300
300     ($12C) :  255 ($FF)
301     ($12D) :  255 ($FF)
302     ($12E) :  0   ($00) @
303     ($12F) :  0   ($00) @
304     ($130) :  255 ($FF)
305     ($131) :  255 ($FF)
306     ($132) :  0   ($00) @
307     ($133) :  0   ($00) @
308     ($134) :  255 ($FF)
309     ($135) :  255 ($FF)
310     ($136) :  0   ($00) @
311     ($137) :  0   ($00) @
312     ($138) :  255 ($FF)
313     ($139) :  255 ($FF)
314     ($13A) :  0   ($00) @
315     ($13B) :  0   ($00) @
316     ($13C) :  255 ($FF)
317     ($13D) :  255 ($FF)
318     ($13E) :  0   ($00) @
319     ($13F) :  0   ($00) @
```

```
55  IF C > 127 AND C < 160 THEN  PRINT
     "CTRL-";:C = C - 64
60  PRINT  CHR$ (C): NORMAL :I =
     I + 1: IF  PEEK (37) = 20 THEN
      VTAB 23: PRINT "DO YOU WANT
      MORE LIST ? ";: GET A$: ON
     A$ = "N" GOTO 140: HOME
70  GOTO 40
80 A = 16:B = 1:A$ = ""
90 D =  INT (C / A):A(B) = C - A *
     D:B = B + 1: IF D > (A) THEN
     C = D: GOTO 90
100 A(B) = D: FOR J = B TO 1 STEP
     - 1:A$ = A$ +  MID$ (C$,A(J
     ) + 1,1): NEXT : RETURN
110 L =  LEN (A$):N = 0: FOR I =
     0 TO L - 1:T =  ASC ( MID$ (
     A$,L - I,1)): IF T = 36 THEN
      RETURN
120 T = T - 48: IF T > 9 THEN T =
     T - 7
130 N = N + T * 16 ^ I: NEXT : RETURN
140  HOME : END
```

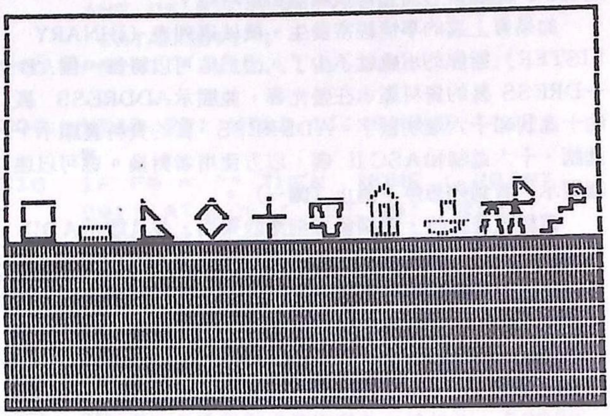
生動活潑的數字迷題

八十年代出生的兒童在接受教育方面肯定比我們的先輩有福。特別是電腦的出現。

這個程式是專針對培養學前兒童對數目的認識學習，程式寓教育於娛樂。程式採用有十個圖形，包括：蛇、狗仔、屋仔、杯子、十字架、鑽石、長方形、四方形、三角形、船仔。電腦會以隨機方式由其中選出一個圖形，再以隨機形式將這個圖形在高解像度螢幕上顯示多次（由1至9次），然後電腦會要求兒童輸入該圖形出現過的次數，若果輸入正確，螢幕會有許多令人感覺有趣的反應出現。

筆者是利用第十二期刊登的一個程式〔圖形建造師〕（見第十二期P.63）來製造這些圖形，各位不妨利用它來繪製更多美麗的圖案。

本文程式共有兩個：COUNTING QUIZ 及 BSHAPES，後者是圖形表，大家可用 BSAVE



BSHAPES，A＄4000，L＄295來存入磁碟。至於程式的輸入方法可參考〔如何輸入機械碼程式〕。

```
  1 REM  ****************************
  3 REM  *    COUNTING    QUIZ     *
  5 REM  *   COPYRIGHT 1985 (C)    *
  6 REM  *   BY COMPUTING AGE      *
  8 REM  ****************************
 80 TEXT : HOME : VTAB 22: PRINT
    "** COPYRIGHT 1985 BY COMPUT
    ING AGE **": GOSUB 1000
100 BL$ = "                    ": REM
    20 SPACES
110 T1$ = "*****************"
120 T2$ = "*               *"
130 VTAB 7: HTAB 12: GOSUB 860
140 PRINT T1$: FOR I = 1 TO 7: HTAB
    12: PRINT T2$: NEXT I: HTAB
    12: PRINT T1$
150 VTAB 9: HTAB 14: GOSUB 860: INVERSE
    : PRINT "COUNTING QUIZ": NORMAL
160 GOSUB 880: PRINT : HTAB 15: PRINT
    "BY": PRINT : HTAB 14: SPEED=
    100: PRINT " COMPUHOUSE ": SPEED=
    255: GOSUB 880: GOSUB 880
170 FOR I = 15 TO 7 STEP - 1: VTAB
    I: HTAB 12: PRINT BL$: FOR J
    = 1 TO 3:BZ = PEEK ( - 163
    36): NEXT J,I: GOSUB 870
180 REM   INTRO
190 HGR : ROT= 0: SCALE= 1: HCOLOR=
    5: FOR I = 2 TO 90: HPLOT 2,
    I TO 277,I: NEXT I
200 HCOLOR= 1: FOR I = 91 TO 157
    : HPLOT 2,I TO 277,I: NEXT I
210 HCOLOR= 3: HPLOT 0,0 TO 279,
    0 TO 279,159 TO 0,159 TO 0,0
    : HPLOT 1,1 TO 278,1 TO 278,
    158 TO 1,158 TO 1,1: HPLOT 2
    ,91 TO 277,91 TO 277,92 TO 2
    ,92
```

```
220  HCOLOR= 0: FOR I = 139 TO 3 STEP
      - 2: HPLOT I,2 TO I,90: HPLOT
     280 - I,2 TO 280 - I,90: NEXT
     I: GOSUB 880
230  FOR S = 10 TO 1 STEP - 1: FOR
     I = 1 TO S: HCOLOR= 3: DRAW
     S AT 27 * I - 19,90: POKE 76
     8,10 * I: POKE 769,5: CALL 7
     70: IF I = S THEN 260
240  IF (S = 10 AND I = 1) OR (S =
     10 AND I = 2) THEN  GOSUB 88
     0
250  HCOLOR= 0: DRAW S AT 27 * I -
     19,90
260  NEXT I,S: FOR I = 1 TO 25
270 C =  INT ( RND (1) * 8): IF C
      = 0 OR C = 4 OR C = 7 GOTO
     270
280 S =  INT ( RND (1) * 10) + 1:
     HCOLOR= 0: DRAW S AT 27 * S
      - 19,90: POKE 768,10 * S: POKE
     769,5: CALL 770: HCOLOR= C: DRAW
     S AT 27 * S - 19,90: NEXT I
290  HCOLOR= 0: DRAW 10 AT 251,90
300  FOR S = 9 TO 1 STEP  - 1: FOR
     I = S TO 10: DRAW S AT 27 *
     I - 19,90
310  HCOLOR= 3: DRAW S AT 27 * I -
     19,90: POKE 768,10 * I: POKE
     769,5: CALL 770: HCOLOR= 0: DRAW
     S AT 27 * I - 19,90
320  NEXT I,S: GOSUB 880: HGR : GOSUB
     880
330  HCOLOR= 3: FOR L = 156 TO 0 STEP
      - 3: HCOLOR= 0: HPLOT 0,L =
     3 TO 279,L = 3 = 1 TO 0,L =
     3: HPLOT 1,L = 1 TO 1,L = 3:
      HPLOT 279,L = 1 TO 279,L =
     3: HPLOT 278,L = 1 TO 278,L =
     3: FOR I = 1 TO 2:BZ =  PEEK
     ( - 16336): NEXT I,L
350  HPLOT 0,1 TO 279,1: HPLOT 0,
     159 TO 279,159: HPLOT 0,158 TO
     279,158: GOSUB 870
360  VTAB 22: SPEED= 100: PRINT "
       PRESS ";: INVERSE : PRINT
     "'RETURN'";: NORMAL : PRINT
     " TO END 'COUNTING QUIZ'  ":
      GOSUB 880: GOSUB 880: SPEED=
     255: HOME : GOTO 910
370  REM  ***** SUBROUTINES *****
390  HOME : VTAB 21: FOR I = 1 TO
     60: PRINT N;" ";:BZ =  PEEK
     (- 16636): NEXT I: GOSUB 86
     0
400  HGR : POKE  - 16302,0: CALL
      - 1998: HOME : FOR I = 1 TO
     10:C =  INT ( RND (1) * 8): IF
     C = 0 OR C = 4 THEN C = C +
     1
410  HCOLOR= C: DRAW S AT 1 +  INT
     ( RND (1) * 255), INT ( RND
     (1) * 170) + 21: POKE 768,20
     * I: POKE 769,20: CALL 770:
     NEXT I
420  FOR J = 10 TO 1 STEP  - 1:C =
      INT ( RND (1) * 8): IF C =
     0 OR C = 4 THEN C = C + 1
430  HCOLOR= C: DRAW S AT 1 +  INT
     ( RND (1) * 258), INT ( RND
     (1) * 170) + 21: POKE 768,20
     * J: POKE 769,20: CALL 770:
     NEXT J
440  FOR I = 1 TO 50: NEXT I: HGR
     : GR :CH =  INT ( RND (1) *
     5 + 1): ON CH GOTO 450,520,5
     60,600,680
450  REM
460  GR : COLOR= 13: FOR I = 39 TO
     0 STEP  - 1: HLIN 0,39 AT I:
      FOR J = 1 TO  INT (I / 2):B
     Z =  PEEK ( - 16336): NEXT J
     ,I
470  FOR J = 19 TO 0 STEP  - 1:C =
      INT ( RND (1) * 15): IF C =
     13 THEN C = C - 1
480  COLOR= C: HLIN 38 - J,J + 1 AT
     38 - J: HLIN 38 - J,J + 1 AT
     J + 1: VLIN 38 - J,J + 1 AT
     38 - J: VLIN 38 - J,J + 1 AT
     J + 1: POKE 768,220 - 2 * J:
      POKE 769,20: CALL 770: NEXT
     J
490  HLIN 0,39 AT 0: HLIN 0,39 AT
     39: VLIN 0,39 AT 0: VLIN 0,3
     9 AT 39
500  GOSUB 870: POKE 768,100: POKE
     769,20: CALL 770: GR : RETURN
520  GR : FOR I = 19 TO 0 STEP  -
     1: COLOR=  INT (15 *  RND (1
     )) + 1: HLIN 0,39 AT 2 * I +
     1: FOR J = 1 TO I:BZ =  PEEK
     ( - 16636): NEXT J,I
530  COLOR= 0: FOR J = 19 TO 0 STEP
      - 1: VLIN 0,39 AT 39 - J: POKE
     768,200 - J: POKE 769,20: CALL
     770: NEXT J
540  GOSUB 860: POKE 768,100: POKE
     769,20: CALL 770: GR : RETURN
560  GR : FOR I = 0 TO 39: COLOR=
     INT ( RND (1) * 15): HLIN 0
     ,39 AT I: POKE 768,5 * I: POKE
     769,20: CALL 770: NEXT I
570  FOR J = 19 TO 0 STEP  - 1: COLOR=
      INT ( RND (1) * 15): VLIN 0
     ,39 AT J: VLIN 0,39 AT 39 -
```

```
     J: POKE 768,143 + 4 * J: POKE
     768,20: CALL 770: NEXT J
580  GOSUB 870: POKE 768,100: POKE
     769,20: CALL 770: GR : RETURN

600  GR : FOR I = 1 TO 40: COLOR=
      INT ( RND (1) * 15) + 1: HLIN
      INT ( RND (1) * 40), INT ( RND
     (1) * 40) AT  INT ( RND (1) *
     40)
610  VLIN  INT ( RND (1) * 40), INT
     ( RND (1) * 40) AT  INT ( RND
     (1) * 40)
620  POKE 768,250: POKE 769,15: CALL
      770: NEXT I: GOSUB 870
630  COLOR= 0: FOR I = 0 TO 38
640 HL(I) =  INT ( RND (1) * 39) +
     1: IF COUNTER(HL(I)) < > 0 GOTO
     640
650 COUNTER(HL(I)) = HL(I): HLIN
     0,39 AT 0: HLIN 0,39 AT HL(I
     ): FOR J = 1 TO 2:BZ =  PEEK
     ( - 16636): NEXT J
660  NEXT I: FOR I = 0 TO 39:COUN
     TER(I) = 0: NEXT I: GR : RETURN

680  HOME : TEXT : VTAB 1: FOR I =
     1 TO 21: PRINT BL$ + BL$: FOR
     J = 1 TO 2:BZ =  PEEK ( - 16
     636): NEXT J,I: FOR I = 1 TO
     23: VTAB I: INVERSE : PRINT
     BL$ + BL$;: POKE 768,2 * I: POKE
     769,5: CALL 770: NEXT I: NORMAL

690  GOSUB 880: NORMAL : FOR I =
     1 TO 125: VTAB  INT ( RND (1
     ) * 23) + 1: HTAB  INT ( RND
     (1) * 40) + 1: PRINT N;:BZ =
      PEEK ( - 16636): NEXT
700  GOSUB 880: FOR I = 23 TO 1 STEP
      - 1: HTAB 1: VTAB I: PRINT
     BL$;: NEXT I: FOR I = 23 TO
     1 STEP  - 1: HTAB 21: VTAB I
     : PRINT BL$;: NEXT I: HTAB 1
     : RETURN
720  FOR I = 1 TO 25:BZ =  PEEK (
      - 16636): NEXT :T = T + 1: IF
     T < 2 THEN  RETURN
730  TEXT : HOME : VTAB 10: HTAB
     5: PRINT "YOU'RE GUESSING, A
     REN'T YOU?": PRINT
740  FOR I = 1 TO 50:BZ =  PEEK (
      - 16336): NEXT I: GOSUB 860
     : HTAB 13: PRINT "TRY ANOTHE
     R": GOSUB 880: GOSUB 880: RETURN

750  HTAB 13: PRINT "TRY ANOTHER"
     : GOSUB 880: GOSUB 880: RETURN

760 S$ = "SQUARES": RETURN
770 S$ = "RECTANGLES": RETURN
780 S$ = "TRIANGLES": RETURN
790 S$ = "DIAMONDS": RETURN
800 S$ = "CROSSES": RETURN
810 S$ = "CUPS": RETURN
820 S$ = "HOUSES": RETURN
830 S$ = "BOATS": RETURN
840 S$ = "DOGS": RETURN
850 S$ = "SNAKES": RETURN
860  FOR DLY = 1 TO 200: NEXT DLY
     : RETURN
870  FOR DLY = 1 TO 500: NEXT DLY
     : RETURN
880  FOR DLY = 1 TO 1000: NEXT DL
     Y: RETURN
900  HGR : HCOLOR= 3: IF T = 2 THEN
     T = 0
910 S =  INT ( RND (1) * 10) + 1:
      ON S GOSUB 760,770,780,790,
     800,810,820,830,840,850
920 N =  INT ( RND (1) * 10)
930  HPLOT 0,0 TO 279,0 TO 279,15
     9 TO 0,159 TO 0,0: HPLOT 1,1
      TO 278,1 TO 278,158 TO 1,15
     8 TO 1,1
940  FOR I = 1 TO N: IF N = 0 GOTO
     970
950 C =  INT ( RND (1) * 8): IF C
      = 0 OR C = 4 THEN C = C + 1

960  HCOLOR= C: DRAW S AT X(I),Y(
     I): POKE 768,200: POKE 769,1
     5: CALL 770: GOSUB 860: NEXT
     I
970  POKE  - 16368,0: VTAB 22: PRINT
     "HOW MANY ";S$;" ? ";: GET B
     $: PRINT B$: IF  ASC (B$) =
     13 THEN  TEXT : HOME : END
980  IF N <  >  VAL (B$) OR ( ASC
     (B$) < 48 OR  ASC (B$) > 57)
      THEN  GOSUB 720: ON T GOTO
     970,900
990  GOSUB 390: GOTO 900
1000  REM  INITIALIZE
1010  PRINT  CHR$ (4);"BLOAD BSHA
     PES"
1020  POKE 232,0: POKE 233,64
1030  DIM HL(39),COUNTER(39),P(20
     )
1040  REM
1050  FOR I = 0 TO 20: READ P(I):
     POKE 770 + I,P(I): NEXT I
1060  DATA  173,48,192,136,208,5,
     206,1,3,240,9,202,208,245,17
     4,0,3,76,2,3,96
1080  FOR I = 1 TO 9: READ X(I),Y
     (I): NEXT I: RETURN
1090  DATA  30,60,80,60,130,60,18
     0,60,230,60,30,120,80,120,13
     0,120,180,120
```

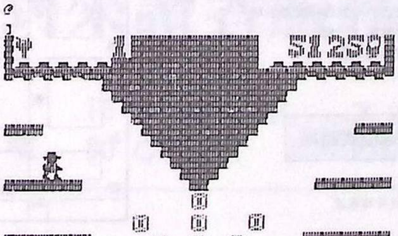
生動活潑的數字迷題

```
*****************
*    BSHAPE     *
* A$4000,L$295  *
*****************
4000- 0A 00 16 00 4F 00 76 00
4008- AC 00 E5 00 07 01 4D 01
4010- A8 01 F1 01 5E 02 2D 2D
4018- 2D 2D 2D 2D 2D 25 24 24
4020- 24 24 24 24 24 3F 3F 3F
4028- 3F 3F 3F 3F 37 36 36 36
4030- 36 36 36 2E 2D 2D 2D 2D
4038- 2D 2D 25 24 24 24 24 24
4040- 24 3F 3F 3F 3F 3F 3F 37
4048- 36 36 36 36 36 36 00 2D
4050- 2D 2D 2D 2D 2D 2D 25 24
4058- 24 3C 3F 3F 3F 3F 3F 3F
4060- 3F 36 36 2E 2D 2D 2D 2D
4068- 2D 2D 25 24 3C 3F 3F 3F
4070- 3F 3F 3F 36 36 00 2D 2D
4078- 2D 2D 2D 2D 2D 2D 3C 3C
4080- 3C 3C 3C 3C 3C 3C 3C 3C
4088- 3C 3C 3C 3C 34 36 36 36
4090- 36 36 36 2E 2D 2D 2D 2D
4098- 2D 25 27 27 27 27 27 27
40A0- 27 27 27 27 27 36 36 36
40A8- 36 36 36 00 49 49 09 2C
40B0- 2C 2C 2C 2C 2C 2C 3C 3C
40B8- 3C 3C 3C 3C 3C 3C 3E 3E
40C0- 3E 3E 3E 3E 3E 2E 2E 2E
40C8- 2E 2E 2E 2E 2E 24 25 25
40D0- 25 25 25 24 27 27 27 27
40D8- 3F 37 37 37 37 37 36 27
40E0- 35 35 35 35 00 49 49 21
40E8- 24 24 24 3F 3F 3F 27 2D
40F0- 2D 2D 25 24 24 24 35 36
40F8- 36 36 2D 2D 2D 35 3F 3F
4100- 3F 37 36 36 36 07 00 49
4108- 49 49 49 21 24 24 25 24
4110- 24 25 24 3C 3F 3F 3F 3F
4118- 3F 3E 3F 37 36 36 36 36
4120- 2D 2D 2D 3C 24 24 3C 24
4128- 3F 37 36 36 36 2D 2D 35
4130- 36 36 2D 2D 2D 24 24 2C
4138- 24 24 2C 24 3C 3F 3F 3F
4140- 3F 37 35 36 35 36 2E 36
4148- 36 2D 2D 05 00 29 2D 2D
4150- 2D 2D 2D 2D 24 24 24 24
4158- 24 24 2C 27 27 27 27 27
4160- 27 27 37 37 37 37 37 37
4168- 37 2F 36 36 36 36 36 36
4170- 2D 2D 24 24 24 24 24 2C
4178- 2D 2D 36 36 36 36 36 2E
4180- 2D 24 24 24 24 24 24 3C
4188- 3C 3C 3C 3C 3F 3E 3E 3E
4190- 3E 36 36 36 36 36 36 2E
4198- 2D 25 24 24 24 24 2C 2D
41A0- 36 36 36 36 36 3F 07 00
41A8- 49 49 2D 2D 25 2D 25 2D
41B0- 2C 2C 2C 2C 3F 3F 27 24
41B8- 24 24 24 24 37 36 3E 3F
41C0- 3F 37 36 36 36 2D 2D 2D
41C8- 24 24 24 3F 3F 37 36 36
41D0- 3E 3F 3F 3F 3F 36 35 2E
41D8- 2E 2D 2E 2C 2D 2D 2C 2D
41E0- 2C 25 25 3F 3F 3F 3F 3F
41E8- 3F 3F 3F 36 35 35 2C 35
41F0- 00 21 24 25 24 25 24 25
41F8- 24 35 36 35 36 35 36 35
4200- 36 25 24 27 24 27 24 27
4208- 2C 2D 2D 2D 35 36 3E 36
4210- 36 37 36 25 24 25 24 2C
4218- 24 2C 36 2E 36 36 35 36
4220- 25 24 27 24 3C 24 27 24
4228- 2D 2D 2D 25 24 27 27 27
4230- 27 27 3F 36 6E 1F 36 36
4238- 36 2D 2D 2D 25 25 24 27
4240- 27 27 27 27 27 3F 37 36
4248- 36 36 36 3E 3C 3F 3F 3F
4250- 3F 3F 3F 37 2D 35 3E 3E
4258- 36 37 36 37 2E 00 2D 2D
4260- 25 24 2C 2D 2D 24 24 2D
4268- 2D 25 24 2C 2D 2D 24 24
4270- 27 3F 3F 36 36 36 2D 2D
4278- 2D 24 24 3C 3C 3C 3F 3F
4280- 36 2E 0D 15 DF 33 36 36
```

戰無不勝的地底戰士

傻豹

這隻新GAME 其遊戲形式類似MINEAR 2049，但其娛樂性就較MIN—EAR 2049爲高，這隻GAME 的玩法是取鎖匙將各間房的門開啓，然後找尋出路以便通到下一LEVEL ，各房間的門都要以其相同顏色的鎖匙才可開啓。小弟曾經算過每一LEVEL 內有95間房，因此要在95間房中找尋一條出路眞是相當困難，特別是到達迷宮黑暗的底層時必須先找到火把方可進入，而每一間房更有各色各樣的機關及守護神阻止你取鎖匙及找尋通道。對於這高難度的GAME 要過一個LEVEL 已是相當難的事，因此要順利找尋通道，通到下一LEVEL 的最好方法就是將人數（機會）增加，而所須的工具就是WILD CARD及RAM CARD各一塊。可能各位讀者會覺得奇怪爲何小弟每次都是用WILD CARD 來修改玩者的機會，而不去用其他無須WILD CARD的方法去



地底戰士的其中一

修改機會，原因是在高登上市的GAME ，其程式的加鎖方法眞是千變萬化，所謂千變萬化是指各商店將一些新GAME 加上自己的鎖，因此同一隻GAME 可能有不同的鎖，因此要指導讀者直接去修改磁碟內的資料是相當困難的事。這個亦是電腦時代所刊出的抄碟方法，對自己的碟行不通的原因。

而用WILD CARD 去修改資料的好處是GAME 的內容很少人去修改，因爲要修改其內容是相當費功夫而且無價値。而 GAME 內資料的所在位置是互通的，因此修改資料的最好方法就是用WILD CARD去修改。

修改地底戰士之前先將其LOAD好，然後按〔RETURN〕去開始遊戲，當遊戲開始後，便照下面修改：

①按WILD CARD上的鍵
②按〔RETURN〕
③按M
④按C089 C089〔RETURN〕
⑤按D0E0 ：7E〔RETURN〕
⑥按C050 C057 C052 B77 G〔RETURN〕

當127 次的機會接近用完時，可依照上面的方法修改，可以再次獲得127次的機會。■

各期程式磁碟磁帶總目錄

1
1 突破傳統的鍵盤
2 撞球遊戲
3 勇闖電腦室
4 迷宮救美
5 將 APPLESOFT 變爲 FORTRAN
6 上、下、左、右移動文字畫面
7 測試工具— BOOLEAN 程式
8 騰空更多磁碟空間 一搬移HELLO
9 重現刪除之檔案
10 電腦時代月曆
11 COMPUTEXT 文字處理
12 時日相距計算程式
13 增強英文字滙能力
14 幼童數學四則運算練習
15 過四關

2
1 貨櫃處理
2 沙蟹幻方局
3 巧破密令
4 美術字繪劃器
5 小階字母處理
6 程式轉載器
7 彩色電視／監視器 色彩測試
8 保留遊戲積分記錄
9 數學套件程式
10 磁碟修護工具
11 加送電腦時代月曆 圖像一個

3
1 DOS 的自動同步追踪器
2 動量不滅定律模擬法
3 自製磁碟索引表封套
4 蘋果編曲器
5 幼童認數字初階
6 賽跑遊戲
7 步步追踪
8 DOS KILLER
9 圍棋遊戲
10 巴哈的古典樂章名作
11 旋轉螢幕畫面
12 雜誌目錄分類歸檔系統
13 神經病醫生一神經鬆弛體操

4
(1) DOS 操作指令簡化器
(2) TEXT FILE 有些甚麽內容?
(3) 立體迷宮
(4) 英文串字練習
(5) 如何以機械語言來修補 DOS
(6) 電腦動畫製作原理。
(7) 豐富設計的學前兒童教育程式
(8) 應用數學套件程式(二)
(9) DOS WRITER
(10) 穿梭機哥倫比亞號機
(11) 找回失落了的 STEP 及 TRACE
(12) TRACER控制常式
(13) 將 RAM 卡作爲磁碟機使用
(14) 黑房計時器
(15) 太陽系模擬系統

5
1 私人約會備忘日曆
2 三度空間顯像系統
3 將 CP／M 檔案轉爲 DOS
4 蘋果萬花筒
5 高解像度文字產生器
6 修訂 DOS 操作指令簡化器
7 濃縮圖片儲存空間
8 幼童數學練習——數字咭
9 加數機大登場
10 備忘 MEMO紙
11 光速計數器
12 剩餘磁碟空 SECTOR數目
13 16進制輸入數字鍵盤
14 問題棋盤
15 神偷智破夾萬
16 隔山打牛
17 大空船降落任務

6
(1) 私人會計系統。
(2) APPLESOFT 擴展版本。
(3) GO] [HELLO 的程式
(4) APPLESOFT 偵錯系統。
(5) DOS 指令再修訂版本。
(6) 布倫時鐘。
(7) AMPERSAND 數基轉換器
(8) 完整索引錄。
(9) 砌積木。
(10) 神奇數字方塊。
(11) 亡命飛車。
(12) 迷宮搜秘。
(13) 太空怪侵略戰。
(14) 累加遊戲。
(15) 爲磁碟命名。
(16) 任意定義鍵盤。
(17) 捉蟲訓練問題。

7
(1) 會說話的蘋果
(2) 私人財政預算系統
(3) 私人收入會計系統
(4) 高解像字符繪劃器
(5) 圖像轉載器
(6) 螢幕文本編輯處理器
(7) 濃縮圖片程式修訂加強版本
(8) 蘋果杯大賽車
(9) 十六個難度的食鬼遊戲
(10) 高爾夫球遊戲。
(11) 空戰驚魂
(12) 美麗圖案設計家
(13) 更多色彩
(14) 列印的困擾
(15) 十一條空 SECTOR
(16) 捉蟲訓練
(17) 80字行咭例子程式

8
(一) APPLESOFT滙編器
(二) 最簡單的拆帶程式
(三) 英文打字練習
(四) 編印單個LABEL
(五) 高解像頁處理工具箱
(六) 簡易快速的記憶移動方法
(七) 眞正要動腦筋的小學四則運算
(八) 七海霸王 (DEPTH CHARGE)
(九) 原子黑洞
(十) 蛇之噩夢
(十一) 收銀機
(十二) 改良的閃動訊息
(十三) 兩欄索引顯示
(十四) 自動三角函數曲綫繪劃系統
(十五) 任意鍵盤 (三個)
(十六) 每期捉蟲問題
(十七) 奇妙的 &

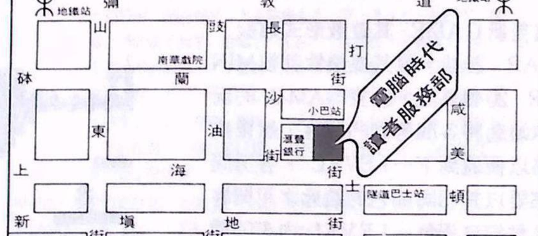
9
(1)磁碟檔案黃頁系統
(2)增強鍵盤輸入及螢幕輸出功能的系統
(3)將工具程式放入DOS佔用的三條磁軌上
(4)快速尋找及磁碟矩陣儲存與提取
(5)新DOS命令改換程式
(6)任意設定目錄索引顯示模式
(7)捉迷藏
(8)古堡之謎
(9)高空擲物
(10)有趣的排序
(11)捉虫問題及答案
(12)第八期捉虫問題答案
(13)印出高解像數目字

10
(1) 可隨意訂定格式之資料紀錄系統
(2) 太空戰士
(3) MARCO鍵盤
(4) 射鴨仔
(5) 奇妙的&——快速分類整理程式
(6) 40軌磁碟工具程式設計
(7) 變數表顯示
(8) 跳越加速器
(9) IQ方塊
(10) 快速檔案搜索
(11) 將DOS移上RAM咭
(12) 培養兒童的記憶能力
(13) 初中代數式運算
(14) 萬用數基轉換器
(15) 十六欄機械碼編印
(16) 捉蟲問題及答案

11
(1) 家庭購物備忘及存貨紀錄系統
(2) 可隨意訂定之資料紀錄系統 (下)
(3) 奇妙的&專欄——矩陣定值
(4) 40軌磁碟工具程式之(二) 單軌 INIT
(5) 兩個自動減省程式佔用之記憶空間應用程式
(6) 對我說
(7) 趣意盈盈的歡樂棒
(8) 英文語句文法分析練習系統
(9) 一行程式示例 (共七個)
(10) 美化程式列表
(11) JACKIES HEAD
(12) 魔域迷宮
(13) 巨蟹求生記
(14) 自動日子記錄
(15) 空SEOCTOR圖表
(16) 捉虫訓練——FACTORIAL!

12
① 家庭開支及個人消費管理系統
② 電子郵箱
③ 40軌程式COPY工具
④ 諾曼第登陸戰 (模擬戰爭思考遊戲)
⑤ 工兵掘地雷
⑥ 新改良蛇之噩夢
⑦ 魔域迷宮 (APPLESOFT 版本)
⑧ 讓九個月大的嬰孩接觸電腦吧!
⑨ 有多項功能的AMPERSAND工具程式
⑩ 磁頭清潔器
⑪ 程式拯救者
⑫ TAB & XY
⑬ 符號逆輸型 (RPN) 計數機
⑭ 比RAM DISK更好的RAM咭控制中心
⑮ 圖形建造師
⑯ 捉虫訓練——SINE曲線
⑰ 一行程式

歡迎補購各期程式磁碟或雜誌



精選程式磁碟

編號：CA00A
內容：遊戲及教育程式
遊戲程式
1、捉球遊戲 (第一期)
2、勇闖電腦室 (第一期)
3、迷宮救美 (第一期)
4、過四關 (第一期)
5、沙蟹幻方局 (第二期)
6、貨櫃處理 (第二期)
7、賽跑遊戲 (第三期)
8、步步追踪 (第三期)
9、圍棋遊戲 (第三期)
10、立體迷宮 (第四期)
11、穿梭機哥倫比亞號 (第四期)
教育程式
12、增強英文滙能力 (第一期)
13、幼童數學四則運算 (第一期)
14、數學套件程式 (第二期)
15、動量不滅定律模擬法 (第三期)
16、幼童認數字初階 (第三期)
17、豐富設計的學前兒童教育程式 (第四期)
18、應用數學套件程式之二 (第四期)

編號：CA00B
內容：應用程式精選
1、電腦時代月曆 (第一期)
2、COMPUTEXT 文字處理 (第一期)
3、美術學繪劃器 (第二期)
4、彩色電視／監視器色彩測試 (第二期)
5、保留遊戲積分紀錄 (第三期)
6、自製磁碟索引封套 (第三期)
7、APPLE 編曲器 (第三期)
8、雜誌目錄分類歸檔系統 (第三期)
9、DOS WRITER (第四期)
10、電腦動畫製作原理 (第四期)
11、將 RAM咭作爲磁碟機使用 (第四期)

編號：CA00 C
內容：私人會計系統
(1) 私人財政預算 (第七期)
(2) 私人收入紀錄 (第七期)
(3) 私人會計程式 (第六期)

編號：CA00D
內容：工具程式集
(1) APPLESOFT程式偵錯系統 (等六期)
(2) 將DOS 移上RAM 咭 (第十期)
(3) MACRO 鍵盤 (第十期)
(4) RAM 咭控制中心 (第十二期)
(5) 程式編輯器 (第十三期)

每期程式磁碟現仍售29.9元；
精選程式磁碟每張售價45元
(除CA00C 只售30元)。

各期程式磁碟總目錄

13

(1) 行政思考決策輔助系統——成功人士必備工具
(2) 全能標貼製作器
(3) 大力水手——POPEYE
(4) 火樹銀花賀聖誕——平安夜
(5) 低解像圖輔助繪劃器
(6) 達爾文進化論的電腦模擬實踐
(7) 程式編輯器
(8) 磁碟檔案地圖
(9) 增加DOS的CATALOG暫停功能
(10) 新游標
(11) 40軌磁碟改動器
(12) 餅形圖
(13) 自己設信紙版頭
(14) 捉虫問題及答案
(15) 一行程式得獎作品

15

(1) 活動電子訊息顯示板
(2) 借貸分期還款計算系統
(3) 麥法基遊戲專欄——DANGER RANGER
(4)「福」臨香江
(5) 將高解像頁倒轉
(6) 機械碼輸入編輯器
(7) 程序分解擴展器
(8) 利用印字機印出多款字形
(9) 檔案密碼鎖
(10) 兩個新DOS指令之一MREAD
(11) 電腦繪圖專欄——立體繪圖的原理示範程式
(12) 氫氧碳化學分子排列結構模擬
(13) 豬仔錢箱
(14) 3—6—9門智遊戲
(15) 超級越軌大賽
(16) 電腦音樂動畫——MONICA
(17) 一行程式入選作品
(18) 捉虫問題及答案

14A

(1) 高解像圖魔術師
(2) 以軟件來達成的音響合成器
(3) SCREEN SAVER
(4) 磁碟度身定造器
(5) DOS 的快速存取
(6) 一行程式得獎作品選
(7) HIDE：將程式隱藏
(8) 英文字義辨識練習
(9) 敎恐龍學習加、減運算
(10) 宇宙城保衛戰（麥法基專欄）
(11) 最後一個立體打井遊戲
(12) 月球一小步，人類一大步
(13) 音量控制
(14) 將APPLESOFT程式轉爲TEXT檔案來編輯
(15) 捉虫訓練——DOS指令的執行

16

(1) 專業室內設計師
(2) 大忙人時間顧問
(3) 學習迷宮產生的原理及自編迷宮遊戲
(4) 使用更方便的BOOT DRIVE 2 軟件
(5) LIST大師
(6) CATALOG HELPER
(7) 40軌磁碟SECTOR 使用情況顯示
(8) 磁碟保護器——COPY VTOC 法
(9) 兩個新DOS 指令 MREAD & MWRT（下）
(10) 爲你的程式配上背景音樂
(11) 成雙成對
(12) 吃不到的葡萄不是酸的 !!
(13) 動態／策略兼備的戰爭遊戲：BEACH HEAD
(14) 一行程式入選作品
(15) 捉虫問題及答案
(16) 組合線性方程式解答系統
(17) 太空侵略者封閉戰CLOSING—ALIEN

14B

DOS 3.3 FORMAT （SIDE A）
(A) 令蘋果發言的最佳工具MOCKING BOARD SPEECH WRITER
(B) 本港郵費計算諮詢系統
CP／M FORMAT （SIDE B）
(C) CP／M的密碼檔案
(D) 程式設計與FORTRAN所帶來的好處（雙精度計算例子）

17

(1) 日文書信文字處理系統
(2) APPLESOFT 全域編輯器
(3) II PLUS 亦可擁有視窗顯示功能
(4) 仿效MOUSE PAINT設計的AP PAINT
(5) 商業發貨單
(6) 把損壞的磁軌鎖住
(7) 配合單軌INIT使用的INIT 40軌
(8) WILDCARD 解拆保護原理透視
(9) 兒童串字
(10) 英文聽力測驗
(11) 金字塔數字堆
(12) 麥法基遊戲專欄：TIME PILOT
(13) 武士忠魂
(14) MUSIC CONSTRUCTION SET 背景音樂問題
(15) GRAPHIC SOLUTION 動畫DEMO
(16) VISICALC 篇——DEMO
(17) 一行程式獲選作品
(18) 第十七期捉虫訓練

補購請到——
上海街438號同珍商業中心11字樓1104室（上海街／登打士街交界）

COMPUHOUSE

歡迎大家到來讀者服務部參觀選購

代理經銷：48 K CPU　16 K RAM 咭　Z 80咭　各類高級磁碟
128 K RAM 咭　80 COL 咭　IBM兼容電腦　PRINTER咭

最新產品：

★CP/M FORMAT 40 (VERSION II)
將CP／M格式磁碟由35軌改成爲40軌

★GPLE使用方法詳析（即將出版）
BEAGLE BROTHER 廠出品的一個強效全域編輯器，我們深入剖析其內在功能並將使用方法編寫成書。

★各種磁帶保護破解天書（即將出版）

★MOCKING BOARD技術應用（即將出版）

★TAKE 1 電影大獎優勝作品選（磁碟版本）

★CHECK SUM軟件

■破解答案（磁碟版本，兩面）
（經已出版）
INSTITUE 及 LUCIFER'S RELAM
豪門恩怨/TRACER SANCTION
每張售20元

電腦時代程式磁碟

每張只售HK$29.90

電腦時代每期都刊登多個精彩的程式，其中必然有你喜歡的。當你讀完文章興緻勃勃的坐下來欲進一步深入了解程式的精彩處時，看見那大堆冗長的列表，就甚麼興趣也沒有了！那麼，你爲何不購買一張「電腦時代程式磁碟」呢！

每期出版的電腦時代所登載的程式，均全部紀錄入一張高品質的磁碟上，並隨雜誌同時出版。爲省卻打入程式的麻煩，徒然浪費你寶貴的時間，免除了花精神捉錯打字的痛苦，今日就請就近的特約經銷處或電腦時代讀者服務部/COMPUHOUSE購備一張。

程式磁碟每張只售港幣 29.90 元。

第18期程式磁碟內容

*All Programs are in Apple version

(1) 配合 PRINT SHOP 使用賀咭信封套印製
(2) 圖像／文字頁顯示選擇器
(3) 股票投資分析系統
(4) 麥法基遊戲專欄：RAIN SHOWER
(5) 書信文件編打系統
(6) 方尖塔爭奪戰
(7) 步步爲營
(8) VISICALC 篇：函數@LOOK UP的使用技巧
(9) 卡通人物專門店：藍精靈
(10) 磁碟 LOCKER
(11) VTOC 編輯器
(12) 利用左右箭咀鍵選擇的 CATALOG HELLO
(13) 從六合彩到編寫程式技巧
(14) 生動活潑認數目遊戲
(15) 使用 BASIC 在高解像畫面玩魔術
(16) 機械碼對照表列印
(17) 一行程式入選作品選
(18) 第十八期捉虫問題

本期程式磁碟雙面灌錄　讀者留意

爲方便讀者購買程式磁碟，特新增香港電視服務站爲我們之特約經銷處；現前往各大地鐵及九廣鐵路電視服務站均有代售。

香港電視服務站

太古城服務站

太古城中心（太古城道）

地鐵站

荃灣服務站	樂富服務站
葵芳服務站	九龍塘服務站
美孚服務站	石硤尾服務站
荔枝角服務站	旺角服務站
長沙灣服務站	油蔴地服務站
觀塘服務站	佐敦服務站
九龍灣服務站	尖沙咀服務站
彩虹服務站	金鐘服務站
黃大仙服務站	中環服務站

九廣鐵路車站

九龍塘服務站	大圍服務站
旺角服務站	沙田服務站

中環

百老滙攝影器材有限公司（第五店）
域多利皇后街3號（中環街市側）
電話：5－235513

亞當影音
士丹利街 28 A（陸羽茶樓側）
電話：5－240723

三聯書店讀者服務中心
香港中環域多利皇后街九號（中環街市側）
電話：5~250102

銅鑼灣

百老滙攝影器材有限公司（第十四店）
軒尼斯道 529號地下（三越百貨公司對面）
電話：5－770181

太古城

百老滙攝影器材有限公司（第七店）
太古城中心第 232店
電話：5－672006

西環

創業雜誌中心
西環屈地街22號地下　明愛中心對面
電話：5-478338

旺角

百老滙攝影器材有限公司　（第 1,3,9,10 店）
彌敦道 758號在下（凱聲戲院側）
電話：3－816931
彌敦道700 號旺角中心第二期地下
電話：3－943827
豉油街 26 號地下（近商務書局）
電話：3－884331
亞皆老街 60 B 地下（世界書局對面）
電話：3－956367

深水埗

百老滙攝影器材有限公司（第十三店）
青山道148 號地下（華聲戲院對面）
電話：3－7295938

尖沙咀

智源書局
金巴利道 27－33號永利大厦 2字樓A座
電話：3 － 678482 3－678414

荔枝角

百老滙攝影器材有限公司（第二店）
美孚新邨第四期百老滙街 79 號（影都戲院側）
電話：3－7424923

觀塘

百老滙攝影器材有限公司（第四、十二店）
物華街 3號地下（皇上皇側）
電話：3－419513
同仁街 35 號地下（大家樂側）
電話：3－449940

荃灣

百老滙攝影器材有限公司（第十一店）
沙咀道 230 A（戴麟趾夫人健康院對面）
電話： 0-4938926

屯門

恒達時電腦公司
屯門華都花園地下商場 8 號室
電話：0-252577

特約經銷商：

教育專業人員協會（佐頓及北角會所）

何必自己動手

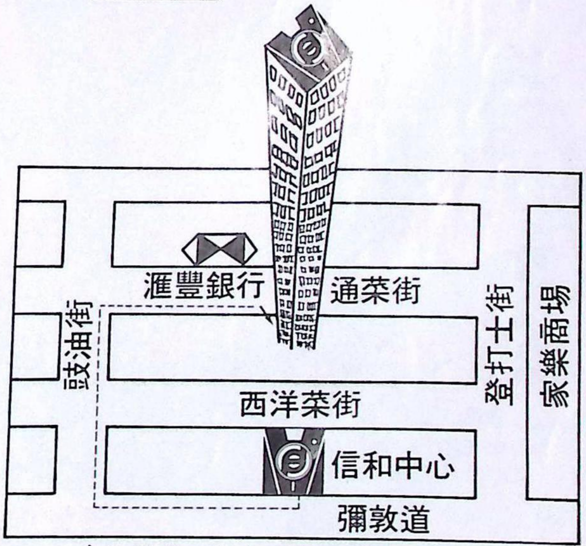
免費 由專業人員上門爲你修理微型電腦



請即加入本公司之保養及免費維修服務

24小時熱線電話服務

3-848129



本公司現已搬往旺角通菜街1A -1J 威達商業大厦19/F，1905室 TEL：3-848129

微型電腦保養維修中心

ROM & RAM computer ltd

OFFICE: Rm. 1905 Witty Coml. Bldg., 1A–1J, Tung Choi St., Kowloon. TEL：3-848129

Telex: 57551 ROMHX

今日最聰明的抉擇

就係選用PARA 磁碟

PARA SYSTEMS

5¼"mini floppy disk



高質素產品　大衆化價錢

日本製造，保証100%無誤差

AUTHORIZED AGENT

SOFT GARDEN

深水埗福華街黃金商場地庫 16B

3-7289066

兒童電腦輔助教育

78 ■ 生動活潑的數字迷題

最新遊戲軟件評介

84 COMPETITION KARATE
THE ISLAND OF BETA
WINNIE THE POOH
SPIDER MAN
HERO

BACK UP 資料館

23 SLOT 0 以外的16K RAM 咭用途
23 用RAM 咭解拆遊戲保護
44 保護磁碟防拷貝方法（WILD CARD的活用）
87 戰無不勝的地底戰士
80 150個LEVEL 的LODE RUNNER

精品店

81 ■ 使用BASIC 在高解像畫面玩魔術
83 ■ 簡單實用的對照性機械碼列印

讀者園地

58 ■ 一行程式參加得獎者
21 ■ 捉蟲訓練
36 誠聘特約撰稿員
18 投稿需知
54 訂閱表格
36 服務部/ AFC 消息

編者言

由第十八期開始，我們將銷售〔電腦時代程式磁碟〕的經銷網進一步擴大，整條地鐵沿線及九廣鐵路多個車站內的香港電視服務站都可以購得我們的磁碟。

編輯部在此要向讀者們道歉，第十七期刊出的讀者調查由於版面的編排失當，許多讀者不捨得將背後的兩頁資料剪下寄來，此點確是出乎我們意料之外，亦足以證明讀者們對我們的愛戴。我們在此希望未填寫問卷的讀者影印一份副本寄來給我們，我們實在很需要這些資料來訂定未來的內容。希望你們合作。同時，爲了公平起見，我們將抽獎日期延至七月，並將得獎者名單刊於第二十期電腦時代。

由於今期稿擠，學校電腦輔助教育專欄暫停一期。

封面攝影設計：梁偉程

特稿

■ PROGRAM SAVED ON DISK

四大磁碟系統磁碟資料存放格式詳析（第二篇CP/M） 13
一張軟磁碟的製造誕生過程 17
■ 從六合彩到編寫程式的技巧（程式編入門技巧之一） 20
■ 圖像/文字頁顯示選擇器 22
ELECTRONIC ART 軟件採用之保護方法分析 59

高級篇

dBASE 篇：編寫 dBASE 程式的第一步 43
■ VISICALC 篇：函數 @LOOK UP 的使用技巧 45

應用軟件庫

■ 書信文件編打器 24
■ 配合PRINT SHOP 使用的賀咭封套製作機 28

中小型工商業應用

■ 股票投資分析系統 30

小型製作專欄

會發聲的鍵盤（精心改良篇） 50

齊齊樂開心坊

■ 方尖塔爭奪戰 62
■ GRAPHIC MAGICIAN 編寫的遊戲 RAIN SHOWER 70
■ 步步爲營（棋類遊戲百種之二） 75

軟件縱橫談

印字機的魔術師：TRIPLE DUMP 32
穿梭蘋果的APPLE MOUSE 和MOUSE PAINT 34

專欄

PRODOS縱橫：UTILITY 磁碟的使用方法 41
機械語言入門：如何控制磁碟機馬達活動（下） 51
■ 卡通人物專門店 可愛的藍精靈 52
麥法基專欄：掀開蘋果高解像畫面的神秘面紗㈢ 55

蘋果工具箱

■ 防止偶然寫花磁碟的 DISK LOCKER 37
■ VTOC 編輯器 39
■ 利用左右箭咀選擇的CATALOG HELLO 19

電腦時代

NO. 18

本刊所登載之各個程式全部適用於APPLE 系列或兼容機種電腦之上

本刊每月廿五號出版。一年十二期。全年訂閱費120 港元，雜誌連程式磁碟爲 475 港元。海外訂戶請參看訂閱收費表。
■出版：電腦時代出版社 ■督印人：伍小華 ■編輯：電腦時代編輯部 ■軟件支援：COMPUHOUSE ■通訊地址：九龍中央郵箱71193 號 ■通訊電話：3-312007 ■ 讀者服務部地址：旺角上海街438號同珍商業中心11字樓1104室（登打士街交界）
■承印者：永恒印刷公司 ■發行人：同德書報社



香港代理：華益企業公司
NO.31-33 KING'S ROAD, 1ST FLOOR,
(NEAR LAU LI ST.) NORTH POINT.
G.P.O. BOX 10617,
HONG KONG

TELEX NO: 75632 PECO HX
CABLE ADDRESS: "PETCOMY" HONG KONG
TELEPHONE: 5-784928, 5-783129
BANKER: HANG LUNG BANK LTD.